THEORETICAL AND EXPERIMENTAL STUDIES ON BASIC RELATIONS BETWEEN REAL WORLD PICTORIAL PATTERNS AND THEIR GENERATING CONSTRAINTS

Michihiko MINOU

Department of Information Science
Kyoto University

November 1982

THEORETICAL AND EXPERIMENTAL STUDIES
ON BASIC RELATIONS
BETWEEN
REAL WORLD PICTORIAL PATTERNS
AND
THEIR GENERATING CONSTRAINTS

Michihiko MINOU

November 1982

Department of Information Science
Kyoto University
Kyoto, 606, JAPAN

Submitted in partial fulfillment
of the requirements for the
degree of
Doctor of Engineering
at
Kyoto University

THEORETICAL AND EXPERIMENTAL STUDIES ON BASIC RELATIONS BETWEEN REAL WORLD PICTORIAL PATTERNS AND THEIR GENERATING CONSTRAINTS

Michihiko MINOU

ABSTRACT

Patterns are considered to be concrete representations of information, so that there must exist some constraints in generating them. The constraints relevant to such processes (which we call "pattern generating constraints") are studied theoretically and experimentally in this thesis. This paper examines the three dimensional depth measurement method and bi-level line drawings.

The depth measurement method developed by the author uses parallel planes of light, each of which flickers in the time domain according to the code word already assigned to it. The code used at the light source was regarded as a kind of pattern generating constraint, and its role in the image processing was demonstrated. Due to this constraint, the depth measurement process became faster and stronger for noise.

Pattern generating constraints in the world of line drawings are considered to originate from the tools used in the generating process. Concretely, these are pens or pencils etc. How those constraints appeared in real world pictorial patterns was investigated, and then a theory was presented. Three kinds of applications, which were concerned not only with image processing but also with digital communication, were tested experimentally in order to verify the theory. The results showed very clearly the advantages of the theory in both the processing efficiency and accuracy.

ACKNOWLEDGMENTS

I would like to express my sincere appreciation to Professor Toshiyuki Sakai for supervising this research, and for his adequate guidance and continuous encouragement.

I am grateful to Associate Professor Takeo Kanade, who is presently connected with Carnegie-Mellon University, for his valuable advice.

I am also grateful to Associate Professor Yu-ichi Ohta, who is presently connected with the University of Tsukuba, for his enlightening discussion with me.

I am indebted to my colleagues in Prof. Sakai's Research Group for their helpful discussions and co-operation. Grateful appreciation is paticularly due to Mr. Shigeyoshi Shimotsuji for his study on MOLD Theory, to Mr. Taka-aki Tatsumi and Mr. Masakazu Ninomiya for their studies in redundancy reduction coding, and to Mr. Yo-ichi Yamamoto for his study in the process of hand-sketched diagrams.

Last but not least, I wish to thank my wife Kyoko for her constant encouragement and assistance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

Pattern is a concrete representation of information. Pattern recognition is one of the popular technical fields used to handle various patterns but it seems to strongly restrict the patterns. In other words, the computer process handles only the patterns favorable to it. We must, therefore, remove these restrictions as much as possible to make the process more useful and practical.

This thesis is concerned with problems of this kind, which are studied in the field of image processing. The key idea is that we are able to take advantage of the constraints which must exist in the pattern generating process.

I-1 New Aspect of Image Processing

Information is said to be an abstract concept in its form without thinking of its meaning. The media used to convey it are physical or chemical materials in the real world. Therefore, an actual process is necessary to put the information onto some material. That part of the material containing some information is called a "real world pattern". As shown in Fig.1-1, real world patterns are fed into computers under certain conditions. Usually a scanning or converting process is applied to obtain image data.

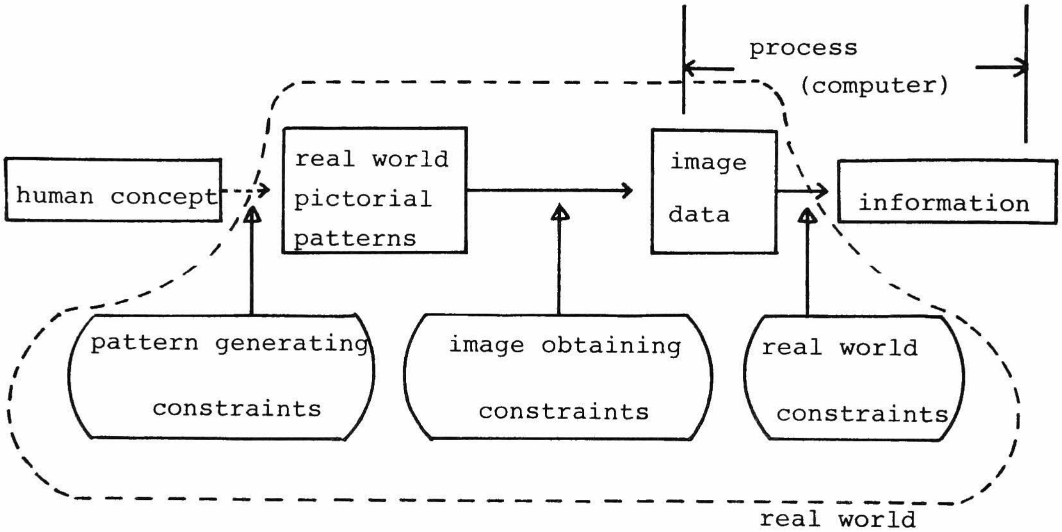


Fig.1-1 Schema of pattern processing.

To consider the general information flow from the original human concept to the computer understanding, there exist three kinds of constraints in each step as shown in Fig.1-1.

(1) constraints to generate patterns
(2) constraints to convert the patterns into computer image data
(3) constraints or knowledge of the real world for the computer understanding process

Various processing techniques have been developed in the field of image processing, but few fully take advantage of the constraints (1). They are mainly concerned with constraints (2) and/or (3).

In this thesis, we consider the basic relations between real world patterns and their generating constraints both theoretically and experimentally. That is, how the pattern generating constraints appear in real world pictorial patterns are theoretically analyzed. Then how computer processes or the processing results are altered by taking advantage of the type (1) constraints is experimentally investigated. Bi-level line drawings and the measurement of depth in three-dimensional space are chosen as suitable fields to verify this idea.

I-2 Three Dimensional Depth Measurement

The first task we introduce in the pattern generating constraints is a depth measurement method. Our method uses parallel planes of light each of which flickers in the

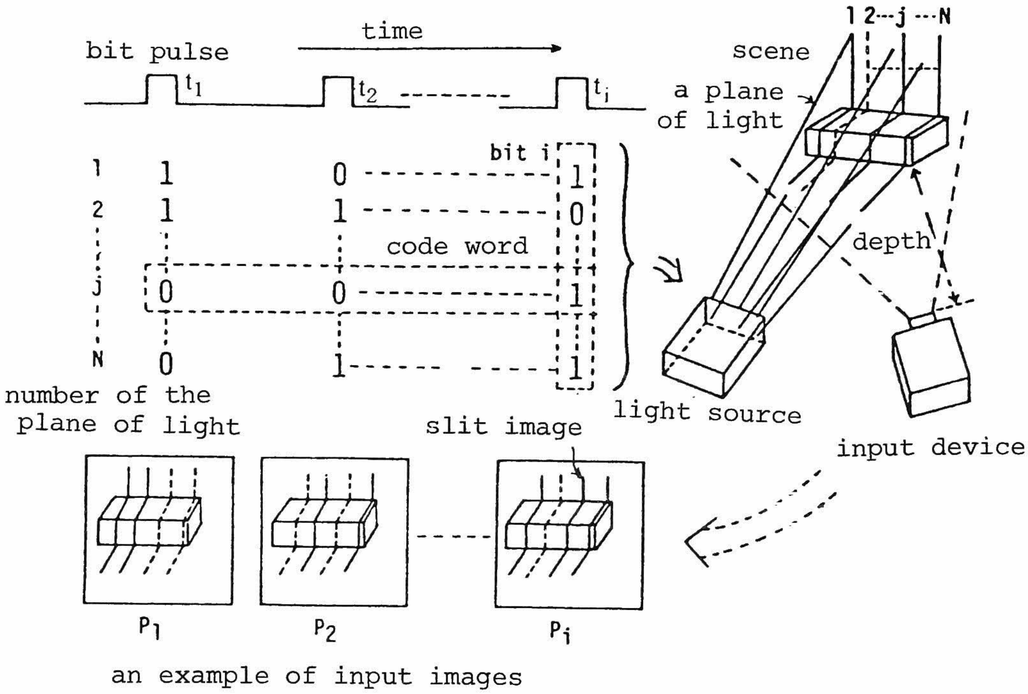


Fig.1-2 Schema of the method of time-coded parallel planes of light.

time domain according to the instantaneous binary code word assigned to it. The concept of our method is illustrated in Fig.1-2. When the code length is n bits, we input n images projected on a certain object in which we locate and "identify" each slit image. Then the depth to the points on the slit images can be calculated by triangulation.

Real world pictorial patterns are considered to be a one-side view of the so called three-dimensional scene. As the slit images on the object are varied with time, the real world patterns in total also become different, although some remain with the same sub-patterns. The pattern generating constraints, then, correspond to the rules generating the planes of light, that is, the code word to be assigned to each plane of light. This is called "light generating constraints". The code words are assigned to derive enough information to identify the planes of light from image data.

In this case, however, the pattern generating process is controlled by the process used to observe and dispose of the generated patterns. It is worth commenting that this case is very similar to an "on-line" system because the computer can utilize all of the information in the pattern generation stage which has already finished.

Our process can take advantage of these light generating constraints, especially in the process of identifying each slit image. Indeed, the other two constraints, image obtaining constraints and real world constraints, are also available, as in conventional depth measurement methods.

One point on which this thesis focuses is the basic relation between real world patterns and their generating constraints. Therefore, the experiments were undertaken to obtain the answer to the question "How is our process different from conventional ones?" From the viewpoint of the SN ratio the experimental results of our method are evaluated. They show that this method, utilizing the generating constraints, is faster and stronger for noise than the conventional method utilizing only image data. The details are described in CHAPTER III.

I-3 Bi-level Line Drawings

The world considered here is that of line drawings generated by the movement of a certain penpoint. In this case, we cannot control the generating process.

The bi-level digital images consisting of line drawings are obtained from the input devices, for example, a digital facsimile. We are going to set a mesh for observation in these images. The patterns divided by the unit mesh are restricted by the fact that the objective world (line drawings) consists of the lines, if the following two conditions are satisfied.

(1) The sampling intervals have two times as fine a resolution as those specified by the

band-limited space sampling theorem.

(2) The widths of the digital lines included are greater than or equal to the size of the unit mesh.

We call these patterns "legal patterns". The schema satisfying the conditions is illustrated in Fig.1-3, and this theory is called "Mesh Oriented Line Drawings Theory (MOLD Theory)". This theory assures that from any kind of line drawings we can construct the model of the original bi-level images, i.e. "LSC-pictures".

Real world patterns, in this case, are the line drawings appearing on a certain paper, so that this world is an analog one mapped by a continuous movement of a writing tool instigated by human concept. It is realized as real world patterns by means of a writing tool.

Thus, the pattern generating constraints only depend on the tools which generate the patterns. The line drawings are considered to be a result of the movements of a writing tool, so that the most important item of these constraints is the width of the lines. As the real world patterns are still in the analog world, a sampling process converting these into digital image data is necessary for computer processing. The image obtaining constraints in this case correspond to the condition (1) above. The condition (2) is an assumption concerning the relationship between the unit of model description and the real world, so that it really has no relation to the constraints.

LSC-pictures, as the results of MOLD Theory, fully include the pattern generating constraints. To investigate the basic relations between real world patterns and their generating constraints, we need experimental evaluation of the descriptive characteristics of LSC-pictures. We investigated the applications of the LSC-pictures model in detail, under the following headings:

(i) adaptive system for noise elimination

(ii) redundancy reduction coding

(iii) reconstruction of hand sketched line drawings

The experimental results show that LSC-pictures are useful and very efficient. The details are described in CHAPTER IV.

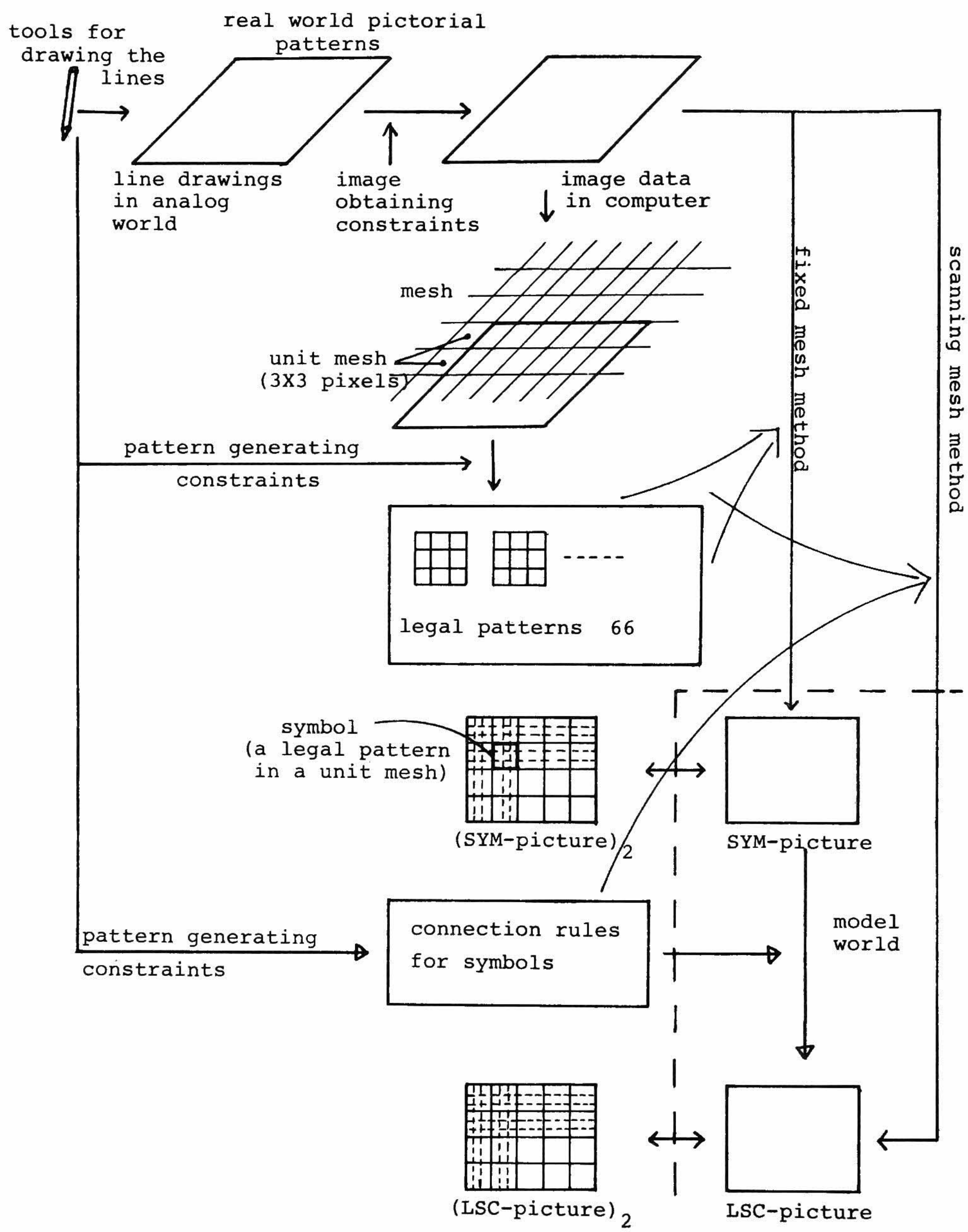


Fig.1-3 Schema of MOLD Theory.

I-4 Related Works

In this section, we present a brief survey in order to provide a perspective to our research. The features are mainly concerned with three kinds of constraints. The works relating to the individual tasks are described in the relevant chapters.

(1) The well known works on interpreting line diagrams as images of three-dimensional scenes, for example, by Huffman [1], Clowes [2] and Waltz [3], are related to our work. They introduced the idea that the lines in an image had their own meanings corresponding to the physical conditions, and the meanings are represented in the forms of labels. Further, for various types of vertices, what combinations of the line labels are permitted to occur in the image under the constraints of the real world are investigated. They try to understand the image by assigning various line labels to each line, which do not cause any contradiction at any vertex. These are very good examples of where advantage is taken of real world constraints.

(2) Another well-known work in computer vision refers to "shape from shading" [4]. The local surface normal is estimated from the amount of reflected light which can be represented by the brightness of the corresponding pixels in the image data. This estimation is only possible under assumptions about the direction of light and about the surface reflection of the object. This kind of research is said to use not only real world constraints but also image obtaining constraints.

(3) The method used to detect the echo signal of radar in noisy circumstances is considered to have close relation to "pattern generating constraints". In this case, the transmitter transmits the radar pulse according to a transmit signal, and the receiver knows this timing. Based on this knowledge, the receiver is able to accumulate the received echo signals measured from the transmitting pulse. Only the echo signal reserving the short term periodicity is accumulated linearly by this process while the noise is averaged to zero. In this way the echo signal is detected [32].

(4) The same idea used in detecting the laser echo is applied to a method of communication. It is called "spread-spectrum communication"[33,34]. At a transmitter, the modulated signal is again modulated by a binary code with a relatively higher bit rate than that of the signal, and this spreads the spectrum. This signal is transmitted in the radio

channel. At the receiver, exactly the same code as used at the transmitter is applied to multiply the entered signal. This is a decoding process using the property of the code. This code is considered to be a pattern generating constraint. The details are described in section III-2.

(5) Jarvis [19] developed a method to detect defects on printed wiring boards. From the observation of a large amount of data, he decided that patterns occurred in 5X5 or 7X7 pixels' window. During investigation of the printed boards, if there occurred any patterns unexpected in a board, it was considered to have an error within it. Only the boards of this kind were selected to be checked in detail. Though the pattern generating constraints are not described explicitly in this case, an example is pointed out which takes advantage of them.

(6) Tominaga et al [35] and Knudson [36] apply the same kind of ideas as ours in order to compress the amount of information needed for communication. They investigated the data through a certain size of window and selected the patterns of high population. They proposed to use only those patterns for transmission, but the selection was arbitrarily done, so that the image quality and the compression ratio are in a trade-off relation. These researches are said to be an implicit application case of pattern generating constraints.

(7) Rosenfeld et al [23] developed a method of noise elimination on multi-level images. They defined patterns in a window and for each pattern they decided what pixels were available to average in order to eliminate noise. By iterating this step, they established the noise elimination process. We consider this research to have taken advantage of the real world constraints.

In this thesis, we aim to use the pattern generating constraints in several tasks. In order to use them efficiently, the problem to be studied is, firstly, what kind of representations are appropriate theoretically and secondly, what possible benefits and advantages should be clarified experimentally by using these constraints.

CHAPTER II

REAL WORLD PICTORIAL PATTERNS AND THEIR GENERATING CONSTRAINTS

II-1 Introduction

Generally, information (message) is considered to be useful or meaningful at the time something is changed, or to have its contents dependent on the human receiver. But, with respect to the sender, it is some abstract concept and usually described by a language. We must represent our concept into actual patterns (e.g. voices or sentences) in order to express it to other people. Various material existing in the natural world can be a medium of information expression. For example, if we use the voice for this purpose, the material is the air and if we use characters, it is a sheet of paper and writing tools. Therefore, it can be said that information is concretely transformed into the form of patterns by means of some material in the real world from a human concept.

Taking this into consideration, any phenomena and any arrangement of things in the real world may be carrying or representing information for some people. We call any part of the real world "real world patterns". For example, wave forms during some period, a photograph taken of some scene, character strings written on paper etc. Especially, those which are represented in visual forms are called "real world pictorial patterns".

Using this terminology, pattern processing is regarded as the process extracting the information from real world patterns. For this kind of machine process, we first put these patterns into a computer through some input devices, expressed as image data. A computer process for this data is applied to obtain the desired results (information).

Hereafter, we restrict the discussion of real world patterns only to pictorial patterns, and consider these steps in more detail. Fig.2-1 shows the whole process from

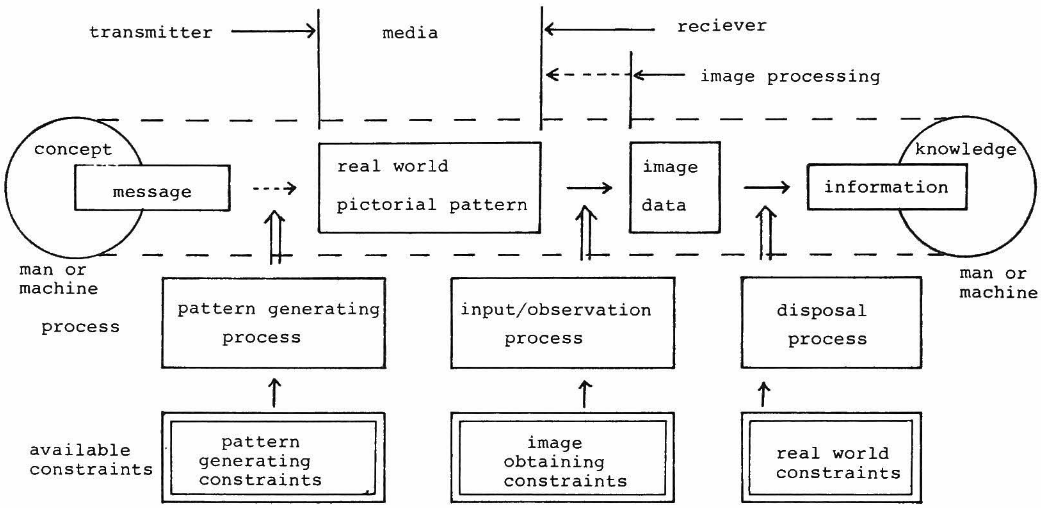


Fig.2-1 Whole steps of pattern processing.

the original patterns to the results useful for information. There are three processes in this figure.

(1) pattern generating process

The origin of the information exists in some material in the real world and the message from the concept and the material realizing the expression together make some real world pictorial patterns. In this process, there may exist some constraints, and we call them "pattern generating constraints".

(2) input process

Real world pictorial patterns are fed into image input devices. There are many constraints (conditions) needed to obtain the image data in the computer. We call them "image obtaining constraints".

(3) disposal process

This process extracts the information contained in the image data, originating from the real world pictorial patterns. The constraints to be utilized are called "real world constraints" in this process.

These three steps are common for information processing, both in the case of men and machines. The first process, i.e. pattern generating process, is done by a person (or a machine); the second, i.e. input process, and the third, i.e. disposal process, are generally done by another person (or machine). This model is considered to be a kind of communication by means of real world patterns, and the latter parts are image processing. We believe that it is important to consider in detail, both in the case of communication and of image processing, more general information processing.

According to the traditional sense, the schema of image processing may be illustrated as in Fig.2-2. That is, as the knowledge about the task, the information

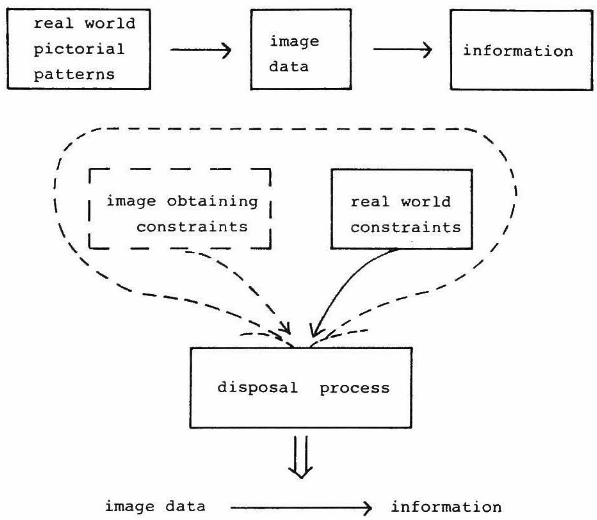


Fig.2-2 Schema of image processing.

processing system in the computer has utilized mainly real world constraints, and in addition, a little research has also utilized image obtaining constraints.

When we consider carefully the schema of pattern disposal, we arrive at the new schema of image processing as shown in Fig.2-3. It may seem strange including more constraints of pattern generating steps to all image processing systems (disposal process), but pattern generating constraints are mainly governed by human society or culture, and

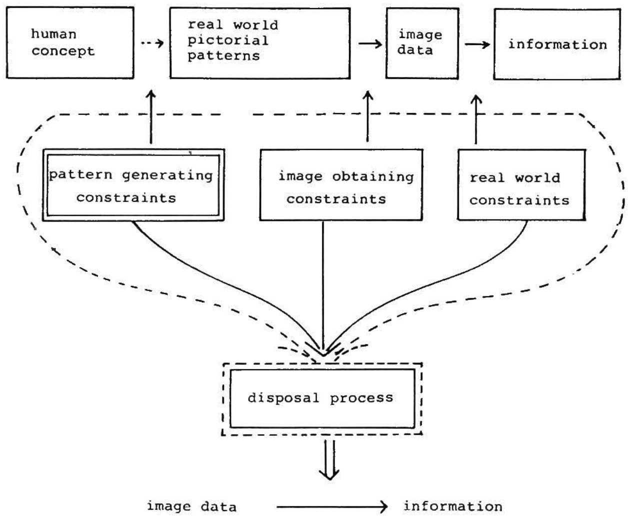


Fig.2-3 New schema of image processing.

superpose themselves on real world patterns. Real world constraints are the rules of natural science. In other words, pattern generating constraints are closely connected with artificial tools, human languages, and documentation engineering.

From this property of pattern generating constraints, it is natural to select objective patterns of man-made things to describe new disposal processes with equal weight. The following sections make these three constraints (steps) in the new schema more explicit.

II-2 Real World Pictorial Patterns

Patterns are considered in their widest sense to be any concrete representation of a human concept. As stated in section II-1, any phenomena, any arrangement of things in the real world will be able to become media for pattern representations. Therefore, all things in the real world are patterns in one sense. Any part or any section of the real world varied for one physical dimension (measure), for example, time or length etc. is called "real world patterns".

A scene projected on our eyes at a certain time, any photographs, any wave forms in the air, LANDSAT data and characters are all real world patterns. Among them, those which are transformed into two dimensional images are especially called "real world pictorial patterns" in this paper.

Generally, the real world may be said to consist of two different kinds of components. One of them is things in nature and the other is man-made things. Accordingly, real world patterns are divided into two categories.

(i) real world patterns of things in nature

Any photograph of natural things, i.e. trees, mountains, sea, sky etc. are in this category. The characteristic property of this category is that patterns are subject only to the laws of nature, i.e. the law of physics, the law of chemistry, or the law of biology found in natural science.

(ii) real world patterns of man-made things

Any picture taken in a factory or in an office etc. or the characters, documents, newspapers, and all the products of industrial activities etc. are included in this category. The real world patterns in this category are subject not only to the law of nature but also to the law of human culture.

The relation of this two categories are illustrated in Fig.2-4. It is clear that the set of the category (ii) is included in the set of the category (i). There is much more needed to process the real world patterns in category (ii) than those in category (i). Our reserch only handles the real world patterns of man-made things.

In this thesis, we treat two kinds of real world pictorial patterns. They are explained as follows:

(1) In the case of three dimensional depth measurement

The object space consists of three components, i.e. objects to be measured, light sources to issue the parallel planes of light and an image input device to input image data into computers. We consider the scene viewed from the position of the image input device in three dimensional space at a moment to be the real world pictorial pattern. Therefore, the origin of the pattern exists in space. Since, the pattern is a view at a moment, it changes with time.

When the light source transmits the parallel planes of light to the object, the real world pictorial patterns then contain the slit images with them. If each plane of light is turned "on" or "off" independently and is synchronized with some clock pulse, real world patterns are changing according to this pulse. That is, if the pulse train consists of n clock pulses, real world patterns are changed n times.

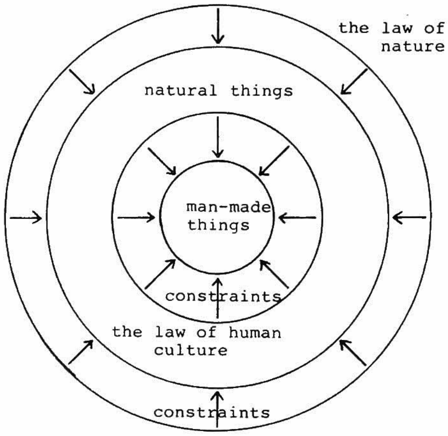


Fig.2-4 Relation between real world patterns of natural things and of man-made things.

(2) In the case of bi-level line drawings

Now let us suppose the objective world consists of all kinds of combinations of lines, that is, characters, graphs, diagrams, charts and figures are the specific names of the real world patterns in this case. These patterns are drawn by some tools, for examples, a pen, a pencil or a fountain pen on paper. Therefore, these patterns are intrinsically bi-level, that is, one color[1)] corresponds to the loci of the tool used to draw the patterns and the other[2)] is the paper itself. These patterns are in an analog world and, indeed, in two dimensional space.

Almost all the documents and diagrams handled in offices or in our usual lives are part of our objective world and they are some of the real world patterns just defined. This world excludes photographs or dither images which have intrinsically multi-level signals. The technique needed to distinguish these areas from documents are already developed [14,15], so that our world covers a very large extent.

These two worlds and their real world patterns are relatively free from strong restrictions compared with those in the conventional pattern recognition task. They are selected by considering only their generating constraints, which are described in the next section.

---

1) Usually black --- the tool is "on" the paper.
2) Usually white --- the tool is "off" the paper so that the background color is the paper itself.

II-3 Pattern Generating Constraints

As shown in Fig.2-4, all the real world patterns are subject to the law of nature. And those of man-made things are, in addition, subject to the laws of the human world, mainly of its culture. For example, a coffee cup's shape is decided not only by the function of retaining the liquid coffee but also by various factors, i.e. design, volume design etc. Concerning the character patterns, their numbers and shapes are closely dependent on human cultures. Real world patterns of man-made things are affected by these two kinds of constraints as shown in Fig.2-5, and they are generally called "real world constraints".

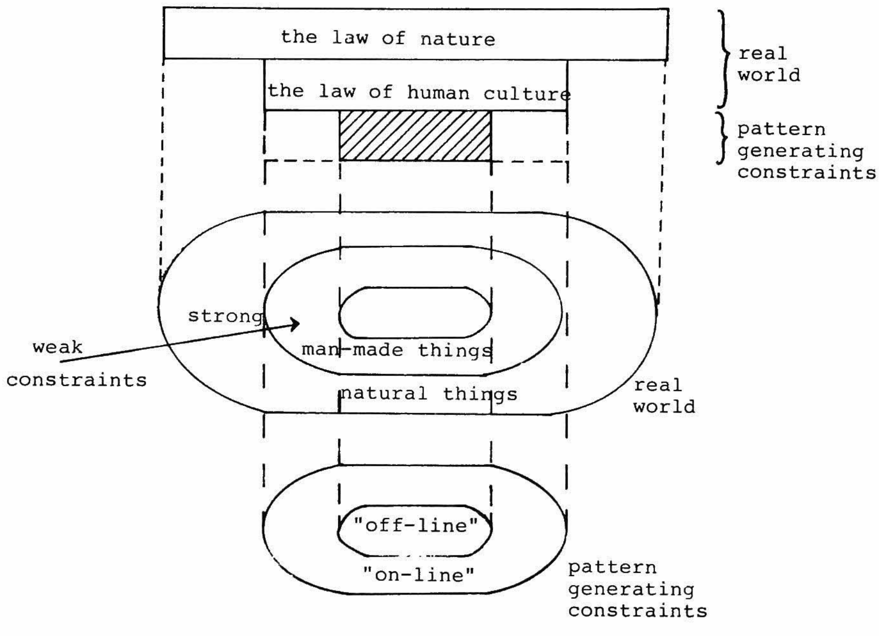


Fig.2-5 Pattern generating constraints.

Since man-made things in the real world have been made by somebody, they may have some extra constraints, not only ones from completely constructed things themselves, but from the process of generating them. We call these kinds of constraints "pattern generating constraints". All the real world patterns are the results of some generating processes, though it may not always be true that pattern generating constraints appear in some form in the corresponding real world pictorial patterns. If they do not appear in the real world patterns, the disposal process cannot take advantage of these constraints, unless it can communicate the generating process. These kinds of constraints are distinguished from the real world ones, and are available to only a part of the real world patterns of man-made things as shown in Fig.2-5.

To utilize these constraints in an actual process, there are two situations based on Fig.2-1. One is the case that the process of generating real world patterns is able to communicate that of processing them. The other is the case where the communication cannot be possible between these two processes.

(i) In the case where all of the pattern generating constraints are available

The situation cited above may be typically a world of a stroboscopic lamp and a camera. That is, when the lamp flashes, the camera takes a photograph of the scene. They are controlled synchronously. The stroboscopic lamp is considered to be a process of generating patterns and the camera serves as a process of disposing of them. Another example is an on-line system, and here we consider the on-line character recognition system. The action of writing characters on to the equipement of the system correspond to the pattern generating process. All the actions can be observed by the system, which makes use of the knowledge as the generated patterns. From these two examples, it is clear that this case is said to have both the process of generating the patterns and of processing, so that they can work synchronously or share premises.

(ii) In the case that only a part of the pattern generating constraints is available

To the contrary, this case is characterized by its off-line property. Only a part of the pattern generating constraints which appear in the real world patterns is available in this case. It is essentially the only knowledge of the tools used in the generating process for the appeared patterns themselves. Character recognition is generally in this case, because the system must handle the characters already written. So that this case is considered to be more difficult than the previous case. To summarize, in this case, the processes of generating the patterns and of processing them are independent of each other. Therefore, only the part of the pattern generating constraints which appears in them is used in the disposal.

The two kinds of concrete research described in this thesis correspond to the cases (i) and (ii), respectively.

(1) In the case of three dimensional depth measurement

Our depth measurement method is an example of case (i), i.e. the knowledge of the pattern generating process can be utilized at the disposal process. The pattern generating process in this case is considered to be the light generating process. Each plane of light has an instantaneous code word. Synchronizing with the bit pulse (timing pulse), all the planes of light should be "on" or "off" according to the bit values (1 or 0) of their code words. Information to distinguish each plane of light is embedded in the form of the code words, and they controll the pattern generating process. Therefore, the concrete pattern generating constraints are considered to be these code words and the bit pulse.

The disposal process knows these constraints and works according to them. That is, the image input device takes n images into the computer successively at the timing of the bit pulse, where n is the length of each code word. The pattern generating process and disposal process are synchronized in the above mentioned sense, and the latter process can readily take advantage of the pattern generating constraints.

(2) In the case of bi-level line drawings

Real world patterns of bi-level line drawings are the collection of the lines drawn on certain paper. The pattern generating constraints here are considered to be those of the tools used in the generating process. Usually, we use a pen or a pencil etc. to draw lines on the paper. We consider the constraints of these tools as the pattern generating constraints. To describe more concretely, they are the spot size of a writing tool and the speed it moves. The former appears in the real world patterns as the width of lines and the latter corresponds to the blur of lines.

Generally, in this case, the pattern generating process has already finished before the disposal process is to be applied. Information appearing only in the period of generating process is almost lost, and only the results contained in the real world patterns are available. Therefore, this example is in case (ii) above, and it may be more difficult to handle these data than those in case (i).

In this thesis, first, the pattern generating constraints appearing in the real world patterns are represented from the theoretical view point. A new theoretical model for bi-level line drawings is proposed based on the width of lines, and concerning some applications, the efficiency of the theory will be tested in various experiments.

II-4 Image Obtaining Constraints

The terminology "image" must first be made clear. It is defined as follows: Image is a two dimensional projection of real world pictorial patterns, in generally N dimensional space, from a certain viewpoint. N is usually 2 or 3. Thus, the image obtaining process is a mapping from N dimensional space to two dimensional space, and only a part of the information contained in the real world pictorial patterns is available in the processing stage of the images. There must be some intended restriction of this mapping process in order to keep as much information as possible in the obtained image. We call these kinds of constraints "image obtaining constraints".

For example, in the case of obtaining the image from three dimensional space, the configurations of the geometrical positions of the input devices, the light sources and the objects are these kinds of constraints. Research [4] to obtain the objects' shapes from their shading information is an excellent example of taking advantage of these constraints effectively. The restriction due to the input devices themselves is also considered to be amongst these constraints.

In our cases, image obtaining constraints are stated in the following:

(1) In the case of three dimensional depth measurement

In the object space there are three dimensional objects which are to be measured. A TV camera is used as an input device to a computer, and light sources are used not only for spacial coded illumination but also for obtaining their generating constraints. The positions of these three components (the objects, the camera and the light source) are considered as the image obtaining constraint in this case.

We measure the depth from TV camera to the objects based on triangulation. If these components are co-linear, the triangulation cannot be applied. This is a constraint on the positions of these three components.

Another constraint is from the positions allowed by the requirement of the disposal process. The disposal process will extract the slit images by scanning the input digital image. A condition that any scanning line of the input image must cross any slit image at most once will provide easier processing. The positions which enable this condition are also image obtaining constraints, and this is described in more detail in CHAPTER III.

(2) In the case of bi-level line drawings

The image obtaining constraints in this case exist in the mapping from two dimensional analog space to a two dimensional digital one. The condition for this process is well known by its name, "sampling theorem", or more clearly in this case band limited sampling theorem.

The constraints in the disposal process are stronger than those specified by that theorem. Since our process handles the digital lines only by specifying their width, the sampling intervals are less than half those in that theorem. This is one of the important image obtaining constraints. Considering the actual characteristics of input devices, the resolution is taken to be unchanged. The sampling constraint is relative to the width of the lines drawn on paper, so that the constraint is only concerned to the width of the lines.

Another important constraint is concerned with the threshold value of the binarization. We use, in experiments, a digital facsimile as one input device into a computer and as the transformation equipement. To do this, at first, the information of the small area around one pixel in the digital image data is derived as a multi-level signal and then the facsimile binarizes this signal by comparing with a threshold value already chosen. In this way, the resultant bi-level image is obtained from the input device.

II-5 Real World Constraints

The disposal process used to handle the input image may utilize knowledge of the constraints on the real world. These constraints are due to the rules of science, and in addition, if the real world consists of man-made things, are due to the constraints of human culture. These are the main constraints discussed and are really very powerful in some cases. Typical research utilizing this kind of constraint is the interpretation of the block world. There is much research on this problem [1,2,3].

Here, we must differentiate very clearly between the real world constraints and real world pattern generating constraints described in section II-3. The difference is regarded as that between the constraints of the things already constructed and those under the process of generation. For example, let us consider character strings drawn on paper.

Pattern generating constraints in this case are, as already stated, the constraints of writing tools. While real world constraints are considered to be the shapes of the characters, and their rows, more explicitly what alphabets and sentences are written. Therefore, pattern generating constraints are very low level or basic constraints, i.e. in signal level. To the contrary, real world constraints are situated at a rather higher level between the knowledge of the task and pattern generating constraints as shown in Fig.2-6.

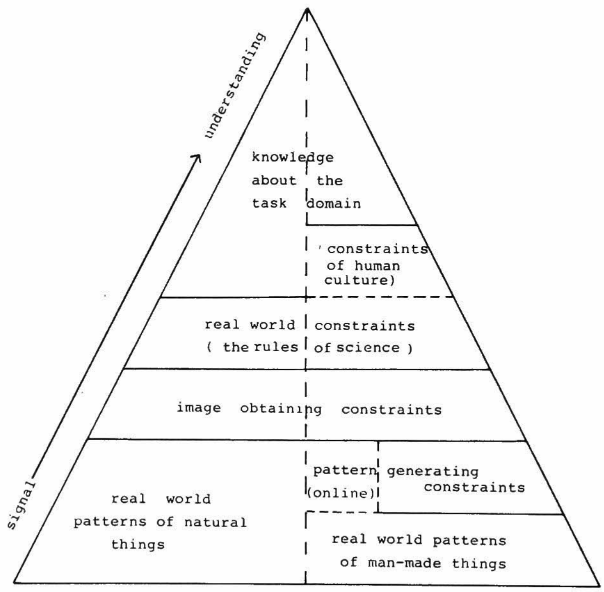


Fig.2-6 Hierarchy of various constraints.

In this thesis, the real world constraints are used in the following forms.

(1) In the case of three dimensional depth measurement

The most important real world constraint in this case is the local continuity of the objects. In our processing of the digital image data, the slit images consist of the pixels with local maximum in the direction of the scanning. The connectivity in the meaning of 8-neighbours must correspond to the conectivity on the objects.

Another assumption about the objects is the uniformity of the surface reflectance, otherwise a slit image cannot be assured to be brighter than the others.

These are the constraints on the objects, i.e. as to the real world itself. To use these constraints efficiently, it may be better to construct a very simple process.

(2) In the case of bi-level line drawings

We make no restriction about the line drawings, so in this case there is no real world constraint available. If we restrict the line drawings to the character strings, the pitch of them or the gaps between character strings are considered to be the real world constraints.

Let us consider the case in which we make an assumption about the relation between the width of digital lines and the size of the unit mesh which is the unit used to describe the real world patterns. This assumption is basic but not a strong assumption on which to construct a theory. The reason the assumption is not strong comes from the fact that we cannot find any relation between this assumption and the real world constraints.

II-6 Basic Relations

The main project in this paper is to investigate the basic relations between real world pictorial patterns and their generating constraints. More generally, the pattern generating process and the disposal process are connected via the real world patterns, and moreover, these two processes are common to both the purpose of information processing and of communication. So that the effect of using the pattern generating constraints in the disposal process is investigated theoretically and experimentally for two different kinds of studies: in the case of three dimensional depth measurement and of bi-level line drawings.

(1) In the case of three dimensional depth measurement

The codes used to controll the light generating process are varied, and their influence on the results is carefully investigated. Experimental results show the considerable advantages of utilizing this constraint, especially in the ability of the process to improve the SN ratio. Owing to this fact, the restriction on the circumstances is very much weakened.

(2) In the case of bi-level line drawings

A theory called MOLD Theory (Mesh Oriented Line Drawings Theory) is constructed by the author which includes the relations on how the pattern generating constraints appear in the real world pictorial patterns. The theory is experimentally tested by examining the real world. The main advantage in using these constraints exists in the fact that the amount of information needed to represent the real world patterns is very much decreased.

To show how practical this theory is, three applications are studied experimentally. From these, we can compare the simplicity, the high speed of the process and results with conventional methods. These are described in CHAPTER IV.

CHAPTER III

A METHOD OF TIME-CODED PARALLEL PLANES OF LIGHT FOR DEPTH MEASUREMENT

III-1 Introduction

Three dimensional object recognition is important in industrial applications or in the field of robotics. The direct measurement of depth, in addition to pictorial brightness, contributes greatly to the recognition process.

One method for measuring the depth in three dimensional space is the "range finding method" based on the triangulation [6]: a plane of light is projected onto the objects from one direction, and a camera from another direction detects its image (which is called "slit image" in this chapter). This method has the restriction that there must be only one slit image in an input image. Thus, the process is simple, but it takes much time for deriving depth because of the scanning of the plane of light.

To speed up the process, we use parallel planes of light. There are several methods using parallel planes of light. Grid Coding method [7] takes advantage of information about the projected light pattern which is regular in the light source. This information gives the orientations of the object surfaces. Unfortunately this method is able to measure only the relative depth, and still worse, the orientation of a surface has two possibilities in principle. Moire method [8] also uses parallel planes of light. A lattice is put between the light source and objects and between the input device and the objects. Moving the lattice at least one pitch and taking a picture by a camera with the shutter open during this time, we obtain the picture of the contour stripe pattern. After the processing of this picture, the depth relations between the neighbouring contours are obtained. If we wish to know the absolute depth in the object space, we must determine the dimension of every contour. This is one of the difficulties of using optical moire systems. These two typical methods implicitly use the assumption that the spacial order of the slit images are the same as that of the planes of light in the light source. (We call it "sequential assumption".) But this assumption is allowable only in very strictly controlled circumstances. That is, it depends on the object shapes, the positions of the light source and the input device. To avoid this condition, each plane of light must be identifiable by itself. In other words, we must "label" the planes of light. There has been research which uses the color for coding [9], but it requires color picture processing, and the number of separable colors is at most 5 in practice.

We have developed a method which uses parallel planes of light that are lit "on" or "off" with time according to a binary code. The idea for this method originally comes from the concept of the method of spread spectrum communication. At the transmitter, the spectrum of the signal is spread by a pseudo-random sequence to a very wide bandwidth, and a receiver, which knows the same code sequence, handles this spread spectrum by using it to decode the original signal. In our case, the transmitter is regarded as the light source which transmits the parallel planes of light, and the assigned code words controll the "on" or "off" of the corresponding plane of light. The receiver in the communication case corresponds to the TV camera in our case and the disposal process which knows the code words can identify the slit images. With them, depth measurement using triangulation becomes possible.

Altschuler et al [12] constructed a hardware system for three dimensional topographic mapping of surfaces. They used M×N discrete beams of a laser as the light sources, each column of them flickered in the time domain. They established point-by-point triangulation as in our principle. Ueda et al [13] developed the same kind of hardware system. They made use of the shutter of PLZT and time sequential coding was applied to it. These two methods aimed to speed up the process of obtaining depth information and they show some evidence towards having practical uses. But here we wish to present explicit expression of basic relations between real world pictorial patterns and their gnenerating constraints.

The code is the key to these processes and generally it is considered to be the pattern generating constraints in our terminology. The generated patterns are the spread spectrum in the case of communication and in our depth measurement method it is the view of the objects with various slit images.

This method is experimentally proved to be faster than the conventional range finding method, due to the parallel illumination and the information about the position of the slit images. Moreover, this method has strong characteristics against noise owing to the pattern generating constraints represented by error correcting code to be used.

We want to get the relation between light generating constraint, (i.e. the code to be used) and the SN ratio of the signal processing stage of the real world patterns. This problem is, we think, one of the most important ones in the field of information processing.

In the next section, we describe the method of spread spectrum communication in order to make the relation more clear between real world patterns and their generating constraints. Almost the same concept is applied in our depth measurement, which is described in section III-3. The most important part of our method is the algorithm to "identify" each slit image using the light generating constraints, i.e. the code to be used at the light source. The details are described in section III-4. Section III-5 presents the explanation of the experiment. The basic relation between real world patterns and their generating constraints in this example is studied from the view point of SN ratio in section III-6.

III-2 Spread Spectrum Communication

III-2-1 Principle

Recently the requirement of more communication capacity and the need for security of the communication led us to the field of spread spectrum communication. It has the properties of allowing sharing of the band width and of noise reduction. The principle of it is not only similar to our approach to image processing, but it is also taken advantage of in our three dimensional depth measurement. Therefore it is described briefly in this section.

There are three general types of spread spectrum technique.

(1) pseudo-random sequence

A carrier is modulated by a digital code sequence having a bit rate much higher than the information signal bandwidth.

(2) frequency hopping

The carrier frequency is shifted in discrete increments in a pattern determined by a digital code sequence.

(3) chirp

A carrier is swept linearly over a wide band of frequencies during a given pulse.

Among these three, we select the pseudo-random sequence to explain the concept

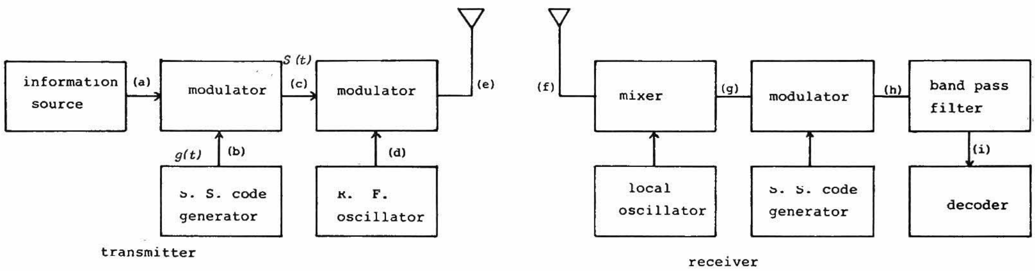


Fig.3-1 Block diagram of a spread spectrum system.

of spread spectrum communication. Fig.3-1 shows a block diagram of a spread spectrum system. At the transmitter, the information is first embedded in a time function, $g(t)$, with a bit rate much higher than the information signal bandwidth. This process spreads the energy of the modulated signal over a bandwidth considerably wider than that of the information signal. The resultant signal is the modulated carrier. It is this signal with wide bandwidth which is transmitted over the radio channel. At the receiver, this signal is multiplied by an exact replica of the spectrum-spreading function, $g(t)$. So that, at the output of the multiplier it becomes $g(t)^2 S(t)$. The spectrum-spreading function, therefore, must be selected on the condition $g(t)^2=1$.

In the case there are n transmitters which use the spectrum-spreading function $g_i(t)$ $i=1,2,$ ---,n, the entered signal of the receiver i is $\sum_j S_j(t)g_j(t)$, and noise term N, the output of the multiplier is then,

$$S_i(t)g_i(t)^2 + \sum_{j \neq i} S_j(t)g_j(t)g_i(t) + Ng_i(t) \text{- - - - - - - - - - - - - - - -} \qquad (3\text{-}1)$$

The family of spectrum-spreading functions must satisfy the following two conditions.

(1) The auto-correlation of any member of the family is

$$g_i(t)^2 = 1$$

(2) The cross-correlation of arbitrary pair of the family is

$$g_i(t)g_j(t) = 0$$

The term (3-1) is, then, ideally as follows.

$$S_i(t) + Ng_i(t) \text{- - - - - - - - - - - - -} \quad \text{- - - - - - - - - - - - - -} \qquad (3\text{-}2)$$

The noise term N is spread by $g_i(t)$, which is one of the reasons why the spread spectrum communication results in noise reduction.

The spectrum of this process is illustrated in Fig.3-2. (a) is the spectrum of the signal to be transmitted. The spectrum is concentrated around the frequency f_i , and has narrow bandwidth. (b) is the spectrum of a spreading function which has a considerably wider bandwidth than that of the signal. (c) is the spectrum of the result of multiplying (a) and (b). (d) is the spectrum of the carrier and (e) is the spectrum of the signal transmitted in the radio channel. So that (a) --- (e) are the results of the processes at the transmitter, while (f) --- (i) are those of the processes of the receiver (see Fig.3-1). (f) is the entered signal with independent interference signals at the frequency f_o . (g) is the spectrum resulting from the removal of the carrier. The interference signal still remains in the spectrum. The next process, which takes the correlation with the spreading function, is the main feature of spread spectrum communication. (h) is the resultant spectrum in which the required signal collapses to its original narrow bandwidth at frequency f_i , while the interference signal is spread. This signal passes through the band pass filter to become the spectrum (i). The interference signal becomes very weak due to the effects of the spreading function and the band pass filter, and the communication may be established even in very bad conditions.

This is the principle of the spread spectrum communication which is a technique able to share a bandwidth and which has a high interference rejection capability. In the next section, we locate this technique from the viewpoint of basic relations between real world patterns and their generating constraints.

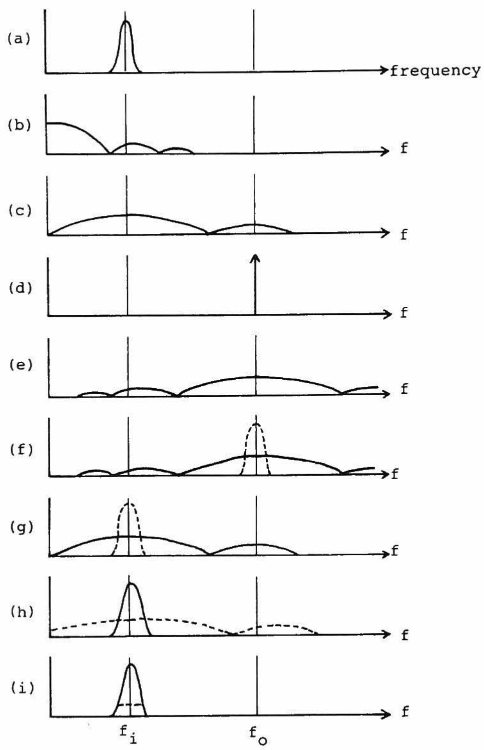


Fig.3-2 Spectrums of spread spectrum processing.

III-2-2 Spread Spectrum Constraints

The key of spread spectrum communication is the spectrum-spreading function which generates the spread spectrum. We present some considerations of this function from our aspect of image processing. The signal transmitted in the radio channel is considered to be a real world pattern in our terminology, though it is not pictorial. It contains the information to be transmitted.

More generally, the spectrum in the radio channel at a certain time is considered to be a real world pattern. It contains various kinds of signals which are generated by various processes. The conventional signals with narrow bandwidth and spread spectrum signals with wider bandwidth are examples. To derive the information from real world patterns, we must know how to generate the patterns. In communication, this is a kind of mutual premises between the transmitter and the receiver.

We may as well call them pattern generating constraints, but those of conventional communication systems are very simple, and the generated spectrum would not be considered to have "structure" in it. To the contrary, the spectrum generated by the spread spectrum communication technique has "structure" due to the spectrum-spreading function. In other words, the spread spectra generated by various spectrum-spreading functions are almost the same in their properties, shapes and bandwidths etc., but the spreading functions can distinguish between them correctly, which we regard as the "structure" of the spectrum. This is why the spread spectrum communication alone is considered to have the pattern generating constraint, concretely the spread spectrum

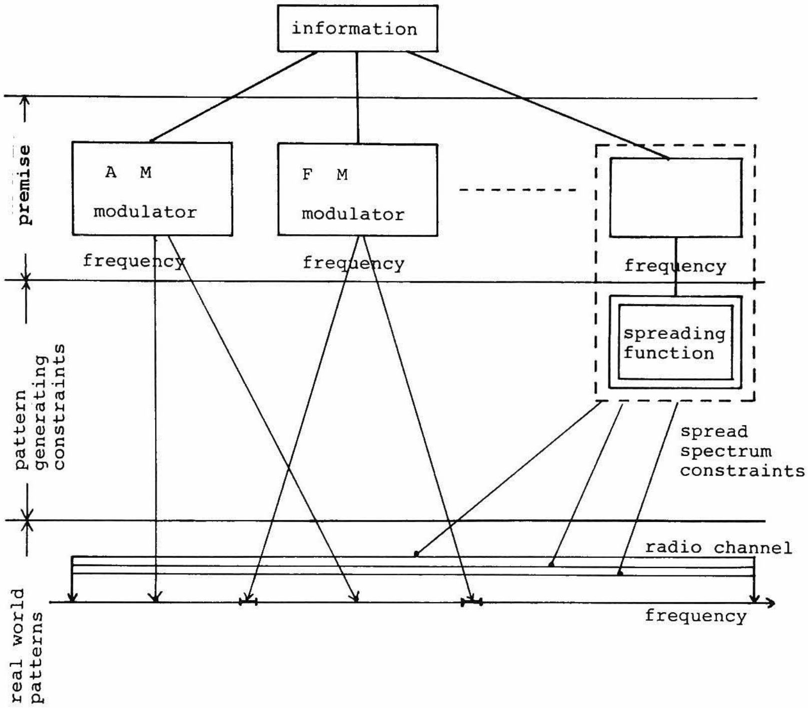


Fig.3-3 Spread spectrum constraints.

constraint (see Fig.3-3).

This constraint is represented in the form of the spectrum-spreading function which must satisfy the conditions (1) and (2) in the previous section. The properties of these functions are very important for the constraint. Indeed, we use other functions as the patten generating constraints and this may not cause the spectrum to spread.

In the next section, we describe the principle of time-coded parallel planes of light, which is a new method for three dimensional depth measurement. This method is originally devised in analogy to the principle of spread spectrum communication, and it has many advantages as to the speed of processing and to noise reduction due to the pattern generating constraints.

III-3 Time-Coded Parallel Planes of Light

III-3-1 Principle

The circumstances (world) of our depth measurement method contain the light source, the image input device and the objects. The light source is regarded as the transmitter and the input device as the receiver. The space between the input device and the light source which contains the objects, corresponds to the radio channel in the case of spread spectrum communication.

We try to use multiple planes of light(=N) parallel in space for making depth measurement in which the method is faster and stronger against noise compared with the

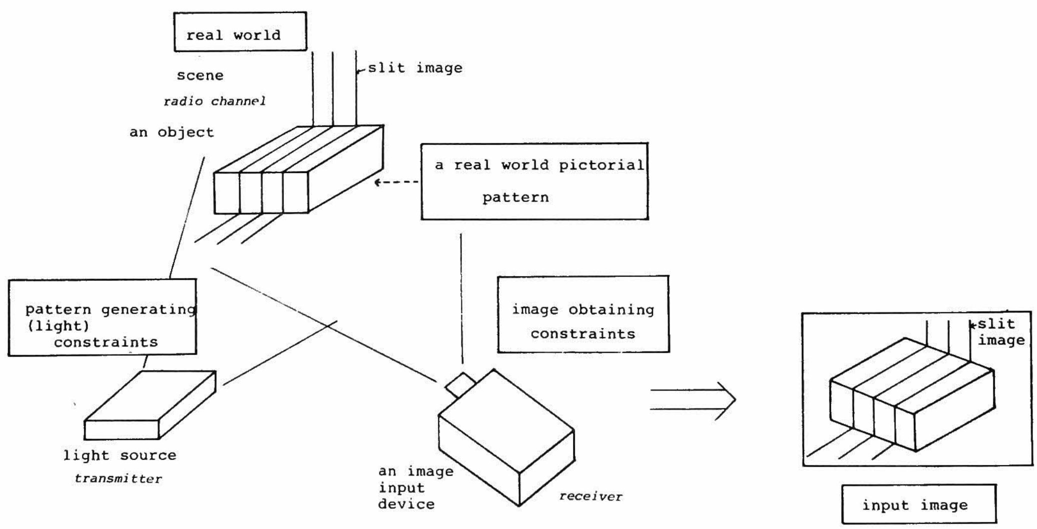


Fig.3-4 Relation between real world pictorial patterns and their generating constraints.

ordinary range finding method. In this case each input image for the scene including an object has many slit images which are considered to be real world pictorial patterns (See Fig.3-4). The main problem is then how to "identify" slit images in the input images. In other words we must solve the problem of finding the "one to one" mapping between the planes of light in the light source and the slit images in the input image. We have to devise a method to "label" each plane of light. Once the mapping (identification) has been found, the depth on the points of the slit images can be calculated by triangulation.

A simple method we attempt here for identification is to give a binary equal length($=n$) code word to each plane of light. This is like the spreading function. At each bit position, individual planes of light are "on" or "off" corresponding to "1" or "0" of the assigned code words and the image is taken. Thus we input multiple($=n$) images

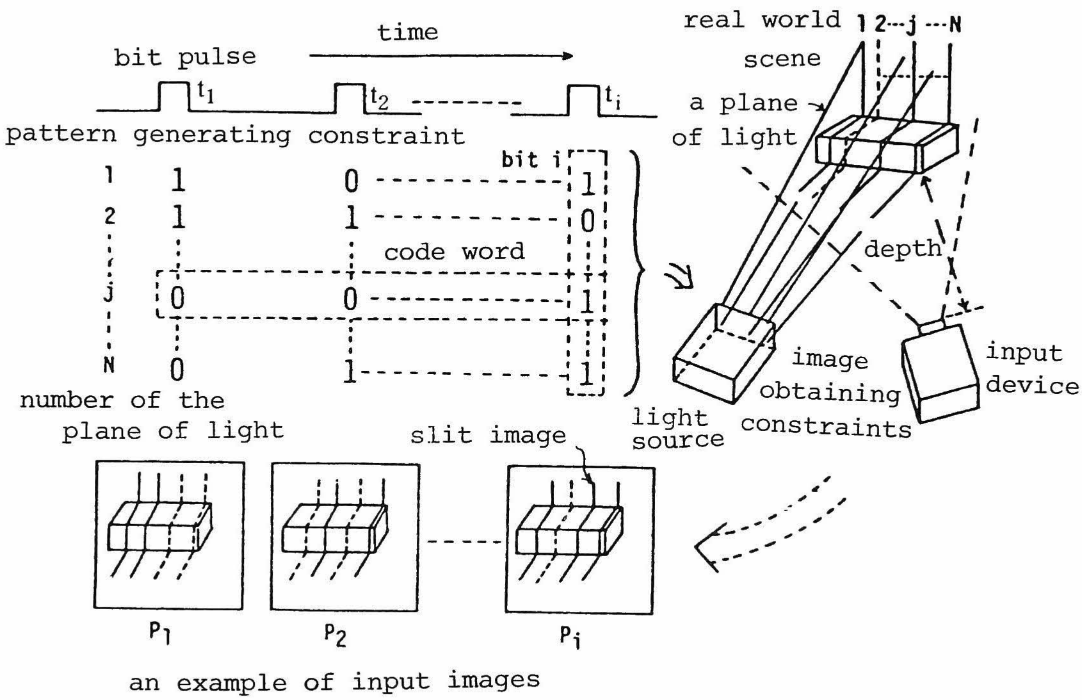


Fig.3-5 Principle of time-coded parallel planes of light.

representing various real world pictorial patterns. This concept is shown in Fig.3-5. At first, we give each plane of light a binary code word as in Fig.3-5, which is considered as the pattern generating constraints, and the on-off of each plane of light is controlled by this code word. All planes of light have the same length code word and are controlled synchronously. To synchronize all the planes of light we introduce a timing pulse called "bit pulse". Suppose that it is time t_i on the bit pulse now. Then all the planes of light are controlled by the i-th bit of the code words. (Code bit "0" means "off" and "1" does "on".) During the same bit time t_i the input camera must take just one image of the scene with the slit images. (See Fig.3-5)

In this way we have a set of n images (P_i $i=1,2,$ — $,n$) for an n bit code to identify each slit image. In a sense, these images are to be one of the "structured" signals which are caused by the light source according to the binary code. The decoding algorithms to identify each slit image work for this "structured" signal i.e. the constraint given at the light source. It is this light generating constraint that makes this method effective against noise. In other words, the constraint is considered to be transformed to the SN ratio. Before describing the algorithms we must mention the restriction on positioning the light source and the input TV camera. This is referred as the image obtaining constraints in CHAPTER II. To make the process for extracting the slit images easier, any row of the digital picture must cross any slit image at most once. As in Fig.3-6, if we install the light

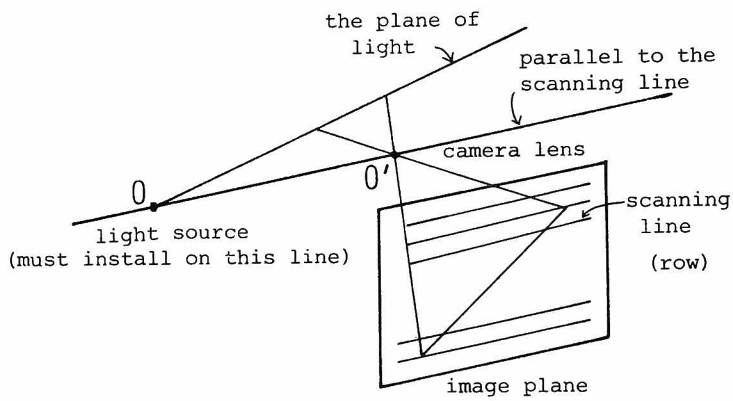


Fig.3-6 Installment of the input device and the light source.

source on the line through the lens center in the input device parallel to the horizontal scanning line of the image plane,[1] our restriction will be satisfied. (This theorem is proved in [10].)

It is worth commenting that there is no restriction on the multiple light sources. Each plane of light has a unique code word so that the system knows a priori which code words are used at a certain light source. This is one of the solutions for the intrinsic problem of triangulation which means that the points seen from the input device but not seen from the light source cannot be measured.

Since we know the positions of the light source and the input device, we can

1) Each scanning line corresponds to each row of the input digital image.

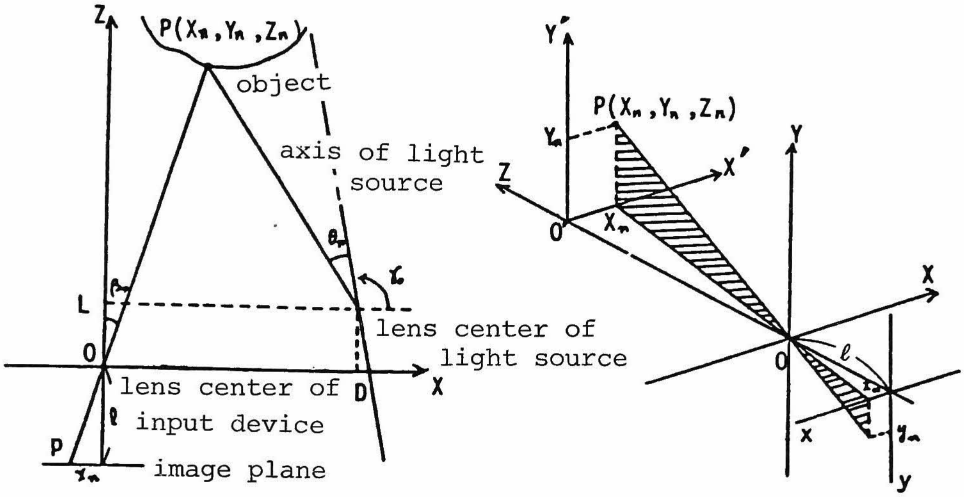


Fig.3-7 System arrangement.

calculate the depth from the image of identified slit images. The relation of these devices and objects are shown in Fig.3-7 with three dimensional coordinates. Simple consideration gives the three equations to obtain X_n, Y_n, Z_n of three dimensional coordinates from x_n, y_n of two dimensional coordinates.

$$\begin{cases} X_n = \{x_n\,(L-D\tan\gamma_n)\}\,/\,\{l-x_n\tan\gamma_n\} \\ Y_n = \{y_n\,(L-D\tan\gamma_n)\}\,/\,\{l-x_n\tan\gamma_n\} \\ Z_n = \{l\,(L-D\tan\gamma_n)\}\,/\,\{l-x_n\tan\gamma_n\} \end{cases} \qquad \text{(3-3)}$$

where L, D, l : parameters shown in Fig.3-7.

III-3-2 Light Generating Constraints

As stated, the real world pictorial pattern is a view of the scene with many slit images and its generating constraint is the code word assigned to each plane of light. A plane of light and the corresponding slit image in the input image make a pair for communication. Therefore, the space between the light source and the input device is in common between n pairs, where n is the number of planes of light.

There arises a problem of "identifying" each slit image in the input images, which is how to find the slit image corresponding to a certain plane of light. To do this, we must make the slit images able to be identified by themselves as in spread spectrum communication. We consider the scene i.e. the objects with slit images as the real world, so that the way to mark each slit image becomes the pattern generating constraint, which has an important role in the disposal process. Here, we use the binary equal length code, and the light is generated according to it.

The code words assigned to the planes of light must satisfy the following condition as in the case of the spreading function.

"The cross correlation between any two code words must be small."

Or in other words,

"The Hamming distance between them must be large."

We can use various kinds of error detecting and correcting codes for this light generating constraint. Our purpose in this chapter is to investigate how the property of the code words, i.e. light generating constraint affects the results, especially the processing speed and the noise reduction capability on the image data. The noise reduction due to the light generating constraint is stated in how much the constraint transforms the SN ratio.

III-4 Algorithms for Identifying Slit Images

Let us describe the algorithms for identifying the slit images with the planes of light from a set of n (equal to the code length) images, and the constraint of the code to be used at the light source. Each input image has information necessary to identify the slit images. Collecting this information with the constraint of the code, the algorithms work to identify each slit image. There are two kinds of constraints the algorithms can make use of. One is the constraint on the code as referred to the light generating one and the other is the information on the spatial positions of the slit images in the input images. We call the former "light generating constraint" and the latter "position information". As we see later, the light generating constraint is useful and important for noise reduction of this, while the position information relates to the processing speed like the synchronization pulse in the field of digital communication.

We propose three algorithms A, B and C for identifying the slit images. As the light generating constraint, algorithm A is of top-down nature (i.e. the code word is known and it is used from the beginning to control the identifying process), and algorithm B and C are of bottom-up ones. On the other side, algorithm A and B use the position information in a bottom-up manner, and algorithm C in a top-down one. The position information is projected to signal level by switching on all the planes of light. So if the objects are moving, this information can not be used at all. In this sense, this information is considered to be a kind of real world constraint.

(1) Algorithm A

The signal representing a slit image appears in the plane of light that is switched by the code word assigned to it. Whether it is or not in each state of the input image depends on that code word. First, we preprocess the input images into binary ones by a thresholding method, in which "1" represents that the pixel is one of the slit images. Suppose that the plane of light j ($j = 1, 2, ---, N$) has the code word $C_j=(c_1^j, c_2^j ---, c_n^j)$ ($c_i^j \in [-1, 1]$), it is transformed by the mapping $[0, 1] \rightarrow [-1, 1]$ for the determination of the threshold level. This algorithm makes the image $B_j(k, l)$ for each plane of light j ($j = 1, 2, ---, N$).

$$B_j(k, l) = \sum_{i=1}^{n} c_i^j P_i'(k, l) \qquad (k, l = 1, 2, \cdots, m)$$

where $P_i'(k, l)$: binary input image

m : spatial resolution of the image

s.t. $P_i'(k, l) = 1$: the pixel is on a slit image

$= 0$: the pixel is not on a slit image

Next, algorithm A identifies the slit image S_j corresponding to the plane of light j ($j = 1, 2, \cdots, N$) in the image $B_j(k, l)$.

$$\exists T_j \text{ such that } T_j = (1/2)\sum_{i=1}^{n}(c_i^j + 1)$$

$$[\text{ideally } T_j = \text{Max } B_j(k, l)]$$

$$S_j = [B_j(k, l) \mid B_j(k,l) \geq T_j \qquad k, l = 1, 2, \cdots, m]$$

The slit images are processed sequentially one by one, therefore the time taken to identify all the slit images is proportional to N (the number of planes of light). Each input image P_i is often processed until the last slit image is processed, and thus large memory capacity is necessary. However algorithm A has a strong noise reduction capability due to the averaging of the input images.

(2) Algorithm B

The algorithm A does not take advantage of the fact that the parallel planes of light are illuminated simultaneously, because the process is sequential on the planes of light. The algorithm B is essentially parallel processing on the planes of light. At first, it extracts the location of the slit images in each input image P_i. The results are accumulated from the first image through the last one. Assume that $C_j = (c_1^j, c_2^j, \cdots, c_n^j)$ ($c_i^j \in [0, 1]$) is the code word of the plane of light j. And $P_i'(k, l)$ is the binary image including the extracted slit images made by one dimensional differential operation. It is accumulated to estimate the image possibility $B(k, l)$ such that ;

$$B(k, l) = \sum_{i=1}^{n} 2^{i-1} P_i'(k, l) \qquad (k, l = 1, 2, \cdots, m)$$

where $P_i'(k, l) = 1$: the pixel in on a slit image

$= 0$: the pixel is not on a slit image

Now, the slit image S_j (j = 1, 2, ---, N) is extracted as follows.

$$\exists T_j \text{ such that } T_j = \sum_{i=1}^{n} 2^{i-1} c_i^j$$
$$S_j = [B(k,l) \mid B(k, l) = T_j \quad k, l = 1, 2, ---, m]$$

That is, the pixels on the slit image S_j have the value equal to the code word C_j.

This algorithm B is much faster than algorithm A because it processes each input image only once. Moreover the necessary memory capacity is small. But the resultant image suffers from much noise.

(3) Algorithm C

In a sense, algorithm B "sees" the whole input image in detail, but the positions at which the slit images in the input images exist are predictable if we first process the image for the case of all the planes of light "on", that is, the image which contains all the possible slit images. (It is exactly the position information.) Algorithm C is designed from this concept. It first decides possible locations of the slit images from that image, (that is, the position information is transformed to signal level,) and in the succeeding images, it only looks at those predicated positions and decides whether there is a slit image or not corresponding to the "1" bit of the code words. Collecting these results through to the last image, the algorithm C identifies the slit images based on the code.

Suppose that the image with all the slit images is $K(k, l)$, which works as a window indicating the necessity of the processing. This algorithm makes use of the equation for $B(k, l)$ as described in algorithm B but the different label of $K(k, l)$.

$$B(k, l) = \sum_{i=1}^{n} 2^{i-1} K(k, l)\, P_i'(k, l) \quad (k, l = 1, 2, ---, m)$$

where $K(k, l)$ = 1 : the pixel is on a slit image

= 0 : the pixel is not on a slit image

P_i' : the same as in algorithm B

In the same fashion as algorithm B, it identifies the slit images S_j ($j = 1, 2, \cdots, N$).

$$\exists T_j \text{ such that } T_j = \sum_{i=1}^{n} 2^{i-1} c_i^j$$
$$S_j = [B(k, l) \mid B(k, l) = T_j \quad k, l = 1, 2, \cdots, m]$$

This algorithm, C, is the fastest of these three algorithms in principle because it makes much use of the position information, and it is efficiently transformed to signal level. It also needs little memory capacity except for the image with all the planes of light "on". However algorithm C is relatively weak against noise. The reason seems to be that the locations of the slit images are extracted from one special image with all planes of light "on".

III-5 Experimental Results

We experiment with two kinds of codes, one is a binary code and the other is a single-error-correcting Hamming code. The code words used in our experiment are shown in Appendix-A. Both codes have 25 planes of light, and the code length of the former is 5 and of the latter is 9. We use a slide projector as a light source and TV camera as an input device. The input image size is 256×256.

An example of input images are shown in Fig.3-8 corresponding to the binary code. This experiment was done in dark conditions, so that the slit images in the input images had high contrast. The abilities of these three algorithms against noise are different in essence, but the results show no distinction (see Appendix-A). This arises since the experimental circumstances are good enough. From the view point of speed, algorithm C is the fastest, and algorithm B is the next. The ratio of their speeds is 1:3:18. The precision of the measured depth is almost the same among these three algorithms. The difference in the processing speed is due to whether the algorithm makes use of the position information or not. Other examples are also presented in Appendix-A.

To show the differences among these algorithms, we conducted experiments under ordinary circumstances, such as under the room fluorescent light in our laboratory. The input images can be seen in Appendix-A. The results of algorithms A, B and C are shown

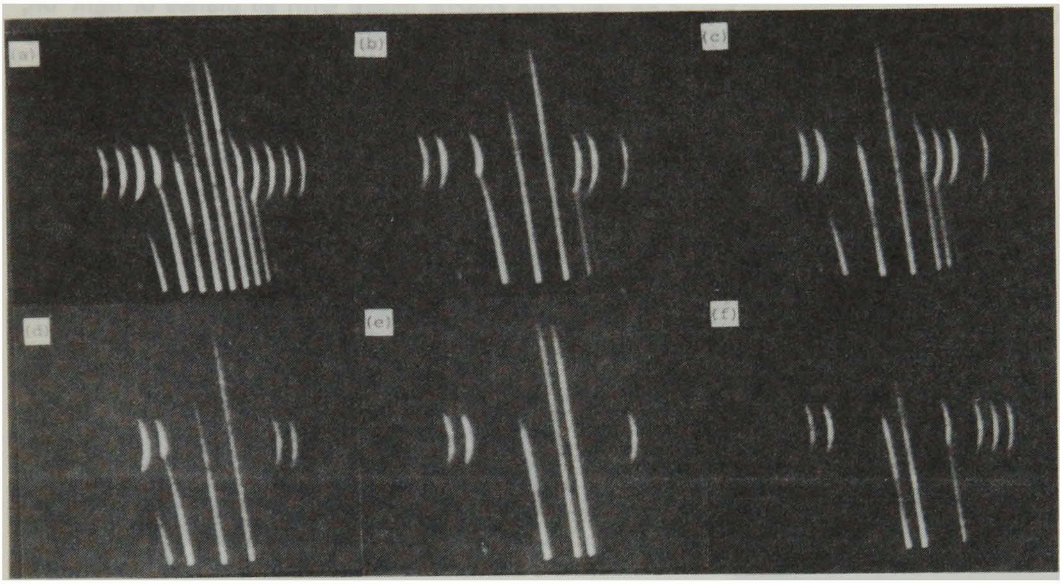
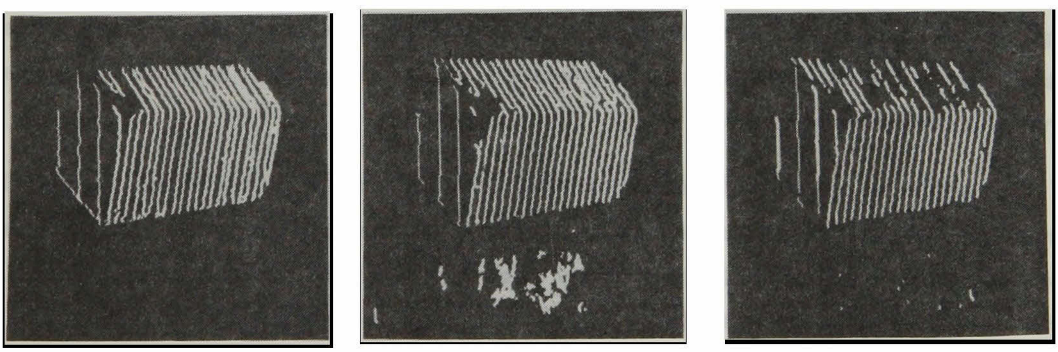


Fig.3-8 An example of input images.



(a) algorithm A (b) algorithm B (c) algorithm C

Fig.3-9 Results of identification (in a bad circumstance).

in Fig.3-9. The order of effectiveness in noise reduction is A, C and B. (the noise appeared because of the reflectance of the table.) Algorithm A averages the input images, so that the locations of slit images are a little notchy, and it makes the precision of the measured depth a little worse. (At most one pixel in the resultant image.) The result of extracting the locations of the slit images using algorithm C is the worst of all. This is because it extracts the locations of the slit images based on only one special image with all planes of light on, whereas the others do from all the input images. The order of the processing speed is C, B and A, and the ratio is 1:3:27. However the speed is dependent on the data, and these values are only an example.

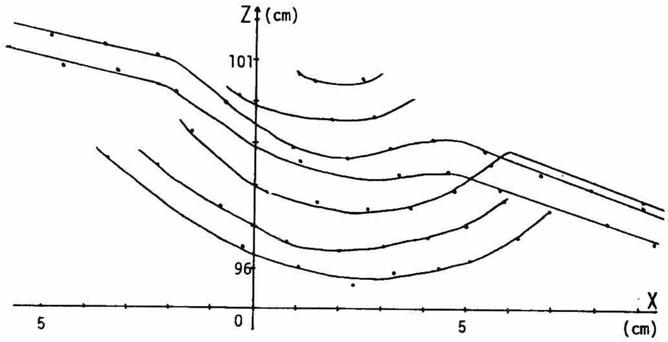


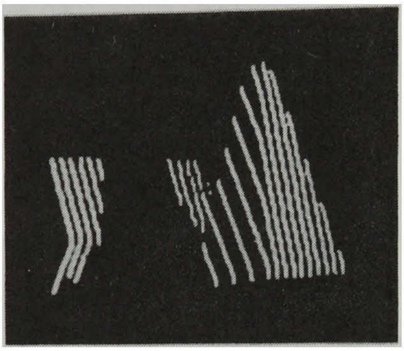
Fig.3-10 An example of measured depth.

After we have an image with identified slit images, we can calculate the depth by equations (3-3). Appropriate cross-sections of the measured object of Fig.3-8 are shown in Fig.3-10 as an example. Generally speaking, the precision of depth is not so good. This is mainly due to the bad estimation of parameters L, D, l. In principle, it is almost the same as that of a conventional method.

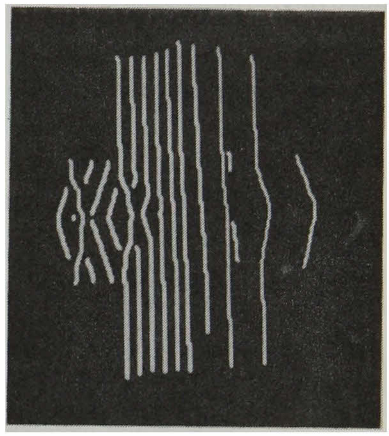
Our depth measurement method can be applied to the case where the sequential assumption is not established, whereas other parallel light methods cannot be applied. This is because each plane of light has a binary code word which can be globally distinct. There is another way to implement this method. If we assume that the sequential assumption is always established, our process will make use of it and improve the processing speed or the noise reduction capabilities.

The property that each plane of light is globally distinct also provides the desirable capability that the above mentioned method is applicable to the case of two or more light sources. We measure the depth in order to know the shape of the object, which is defined locally as the surface orientation, but depth measurement by triangulation includes an intrinsic problem that all the points which are seen from the camera position cannot be illuminated by the light source from another position. If we measure the depth using at least two light sources, a solution of this problem can be presented.

The experimental results are shown in Fig.3-11(a) and Fig.3-11(b), corresponding to the case that the sequential assumption is not applied and the case with two light sources, respectively.



(a) sequential assumption is not established



(b) two light sources

Fig.3-11 Results of identification (in various circumstances).

III-6 Quantitative Evaluation of Light Generating Constraints

In our depth measurement method, the code used at the light source is very relevant to the ability of the system. We can use different codes, depending on the circumstances, in the depth measurement. In noisy conditions, we can exploit the error-correcting capability of the Hamming code with larger minimum distance at the cost of longer code length and, consequently, of longer computation time. In more favorable conditions, a code of shorter length gives faster processing. This is one of the features of our method of depth measurement. In addition, if the assumption that the order of the slit images in the input images may correspond to that of the planes of light in the light source (which we call "sequential assumption") is available, our method will need less code length.

To investigate the code capability for noise reduction and the relation between the light generating constraint and the SN ratio, we applied algorithms B and C, under the same circumstances, using a binary code and a single-error-correcting Hamming code, respectively. The results are shown in Fig.3-12. To specify the abilities quantitatively, we use the terminology of the field of information retrieval. The two ratios, such as "recall ratio" and "relevance ratio", are defined as follows [11]. (See Fig.3-13)

$$\begin{cases} u = F/(F+G) & \text{(recall ratio)} \\ v = F/(F+E) & \text{(relevance ratio)} \end{cases}$$

where F : the number of correct answer(actual)
E : the number of noise pixels
G : the number of missing pixels

F+E represents the pixel numbers of the answer on the slit images in the input images obtained by the actual system and F+G does the pixel numbers of the correct answer which would be obtained by the ideal system.

In our example, there exist rejected pixels (we cannot identify some pixels on the slit images because of noise) and error pixels. These pixels are considered to be independent of "recall ratio" and "relevance ratio", as long as they are small in number. For the results of Fig.3-12, these two ratios, u and v, are calculated from Table 3-1. The error ratio "e" (defined by $\text{error}/(F + \text{error})$) and rejected ratio "w" (defined by $\text{reject}/(F + E + \text{reject})$) are also denoted.

These results show important properties of algorithms B and C in using the redundant images. Algorithm B uses these redundant images mainly to increase the recall factor. As algorithm B examines these images carefully, so the missing slit images are

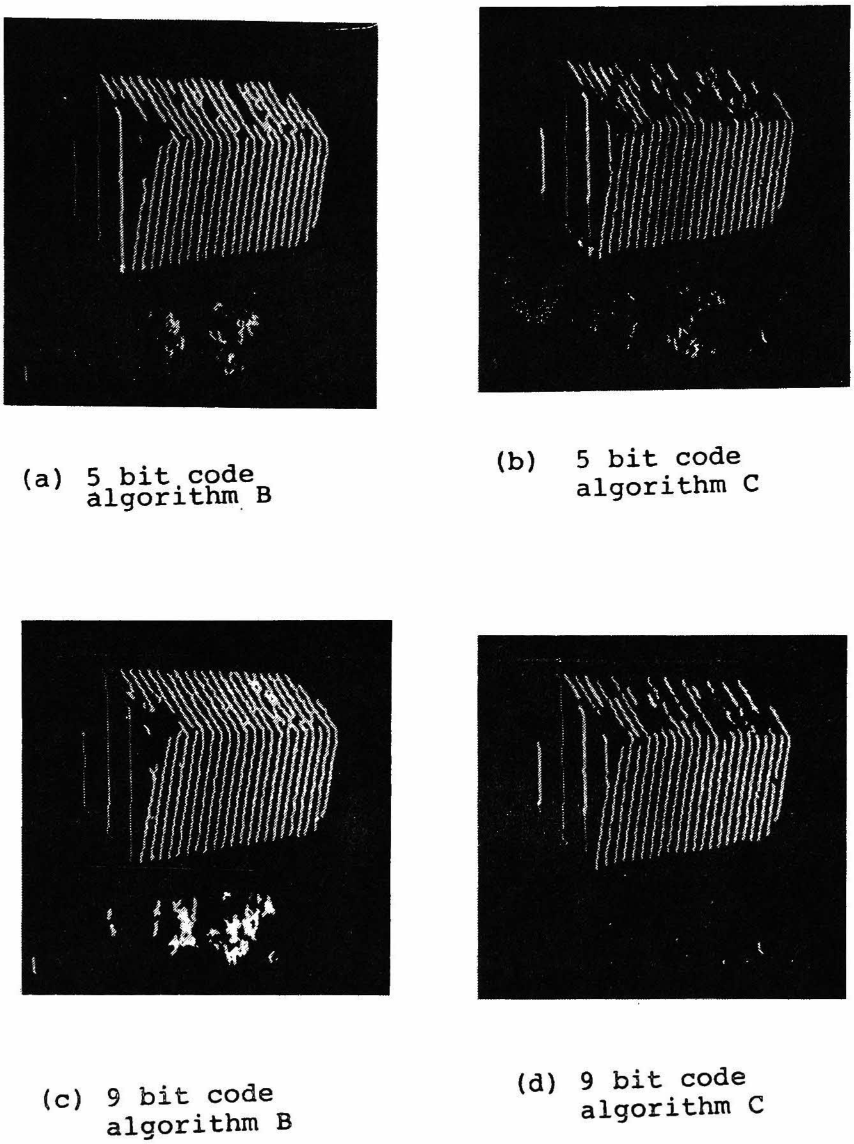
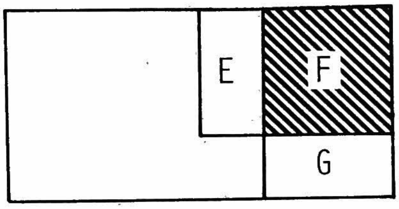


Fig.3-12 Results of identification (with both 5 bit and 9 bit code).

recovered, but at the same time it picks up the noise pixels which prevents increase of the relevance ratio. Algorithm C takes these redundant images to decrease the error ratio because it sees them only through the window derived from the special input image with all planes of light "on". Which algorithm is to be used in a concrete application will depend on the environment and on the following process. In a sense, these ratios are considered



F+E : pixels retreived by the actural system
F+G : pixels retreived by the ideal system
F : received correct pixels
E : noise pixels
G : missing pixels

Fig.3-13 Relation of recall ratio and relevance ratio.

Table 3-1 Recall ratio and relevance ratio etc. for algorithm B and C.

ratio / al-gorithm	number of pixels					ratio (%)			
	F	G	E	reject	error	u	v	w	e
algorithm B 5bit code	2370	181	259	459	41	92.90	90.15	14.86	1.70
algorithm B 9bit code	2875	106	484	29	47	96.44	85.59	0.86	1.61
algorithm C 5bit code	2048	643	232	319	101	76.11	89.82	12.27	4.70
algorithm C 9bit code	2295	479	27	236	1	82.73	98.83	9.23	0.04
algorithm B(1) 9bit code	2494	106	407	410	43	95.92	85.90	12.38	1.69

(1) In this case, the code doesn't have the error correcting capability.

to represent the characteristics of signal against noise[1]. If we do not use the code with an error-correcting capability (processed by algorithm B where the result is shown in Table 3-1), the difference in ability is between that of a 5 bit code and that of a 9 bit error-correcting code. The error-correcting capability is especially useful for decreasing the reject ratio(w). In spite of the above differences due to algorithms, using redundant images, (in other words, to use the code with error-correcting,) is concluded to be very useful for increasing the SN ratio in the resultant images.

To measure how much one image contributes to the two ratios, u and v, and how the light generating constraint appears in the results, we calculate these ratios at every stage of the process in algorithm B. The result is shown in Fig.3-14. In this experimental circumstance (under the fluorescent light, and with the reflectance of the table), it may be best to extract the slit images from about two or three images. Redundant images are indeed useful for increasing recall ratio but have a disadvantage on the relevance ratio. Deciding the optimal number of images to extract the slit images mainly depends on the circumstances. The remainding images will be used to decrease the error ratio rather than to extract them. These results allow us to improve algorithm C. We extract the locations of the slit images from two or three special images with all the planes of light "on". This process is considered to improve the margin for the position information. (We name this algorithm "C1" and process three special images.) Another method to improve the margin for the position information is to enlarge the window in the direction perpendicular to the slit images. In our installation, we can enlarge the window in the direction of the row of the input digital images. This is called "algorithm C2".

For other data, algorithms B, C, C1 and C2 are applied and the four ratios, such as u, v, w and e, are calculated in Table 3-2. The recall ratio of algorithm C1 is much greater than that of algorithm C and almost the same as that of algorithm B, but the processing time is about twice as much as that of algorithm C and about half as much as algorithm B. While, for algorithm C2, the processing time and the ability are almost the same as those of algorithm C. We know that there are two kinds of margins for the position information. One is useful for the recall ratio as in algorithm C1 and the other is for the relevance ratio as in algorithm C2. In our example, algorithm C is weak for the former margin, so that algorithm C1 is more effective than algorithm C. These results show the importance of the margins in the signal level processing.

---

1) We consider the slit images as signal and all the others as noise. For example, noise includes the reflectance of the table, or the edges of the objects etc.

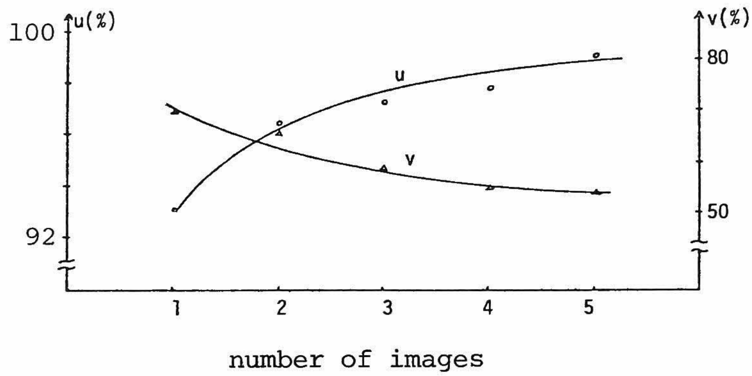


Fig.3-14 Transition of recall ratio and relavance ratio by the number of processed images.

Table 3-2 Recall ratio and relevance ratio etc. for algorithm B, C, C1 and C2.

al- \ ratio gorithm	number of pixels					ratio (%)			
	F	G	E	reject	error	u	v	w	e
algorithm B	1287	41	37	125	0	97.13	97.13	8.62	0.00
algorithm C	698	723	0	32	20	49.22	100	4.38	2.78
algorithm C1	1241	175	1	37	34	87.64	99.92	2.89	2.67
algorithm C2	705	723	0	25	20	49.36	100	3.42	2.76

III-7 Concluding Remarks

In this chapter, we described the depth measurement method by time-coded parallel planes of light. The features of this method are:

(1) Because the number of input images logarithmically increases as the number of planes of light, our method takes less computational time than the conventional range finding method.

(2) Our method is applicable in bad conditions by using codes appropriate for the particular circumstances; noise, object properties, etc.

(3) Each plane of light has a different code word, so we can use this method with the multiple light sources or in circumstances when the sequential assumption is disturbed.

(4) Our method can not be applied for moving objects because the position of the slit images must be kept unchanged through all the input images.

Our depth measurement method can be applied in any circumstances by modifying the code to be used, and is much faster than the conventional range finding method. Moreover this method has the possibility that if we use a feed back loop in a light source, we can calculate the depth depending on the previous measurement. In that case the system can replace the light sources in desirable positions.

We have argued that we can transform the information about the locations of the slit images to the processing speed and that we can also transform the light generating constraint such as error-correcting to the SN ratio in the resultant images. This method shows many advantages, and this is because we can give "structure" such as representing a binary code (i.e. the pattern generating constraint) to the signal of the slit images (i.e. the real world pattern). The problem of noise reduction will become more important in future and we think that the pattern generating constraints will be an important portion of that problem.

CHAPTER IV

MESH ORIENTED LINE DRAWINGS THEORY AND ITS APPLICATIONS

IV-1 Introduction

In this chapter, we handle the world consisting of line drawings. This is an example of the so called off-line case. That is, the disposal process cannot utilize the information of the pattern generating process. The written line drawings are considered to be real world pictorial patterns and they are only links between the pattern generating process and their disposal process. Among the pattern generating constraints, only the part which appears in the real world patterns in a certain form can be available.

The objective world includes any kind of data written by pen-like writing tools and it is intrinsically a bi-level world. It excludes photographs or dither images which originally have multi-level information. For this world, the pattern generating constraints are considered to be information about the tools used in the generating process, and appears in the real world patterns as the width and the blur of lines. The purpose of this chapter is to make the basic relations between real world patterns and their generating constraints into explicit theoretical expressions, and to test these relations by applying them to some actual fields, not only of information processing, but also of digital communication.

In section IV-2, we construct a theory which is called "Mesh Oriented Line Drawings Theory". First, the form of lines in the digital world is defined by their width, and the constraint mapping the lines from the analog world into the digital lines is derived theoretically. (It is exactly an image obtaining constraint). Some experiments show that this constraint is not unlikely to be satisfied in the real world. Next, the assumption about the relation between the width of digital lines and the size of unit mesh, which is taken as an observational unit, is introduced. Based on this assumption, we define "legal patterns" which are assumed to represent a part of the digital lines. The pattern generating constraint in this case is represented theoretically as the form of "legal patterns". This is tested in the real world and we propose new representations of the bi-level line drawings, SYM-picture (picture represented by SYMbols) and LSC-picture (picture represented by Legal Symbol Connections). Each unit mesh in the fixed position must be one of the legal patterns in a SYM-picture, and the fact is derived that there must not exist any illegal patterns in an LSC-picture. MOLD Theory assures that the image satisfying the image obtaining constraint and the assumption can be represented in the form of a SYM-picture and an LSC-picture. For these two representations it has been experimentally proved that less information is necessary compared with the original bi-level image and that the quality of them is not worse than that of the original.

Some applications of MOLD Theory are described in the following sections. In section IV-3, the information processing on a SYM-picture or an LSC-picture is described. The constructed system is an adaptive system for noise elimination. It accepts images of various qualities, and can perform some information processing according to their quality. At first, the system classifies the input images according to their quality into three classes, "blurred" which means the following process must be to increase the black pixels, "stained" which is the contrary of "blurred", and "good quality" which means there is almost no need for noise elimination processing. This classification is based on the statistics of legal patterns in the objective image, and it works very well for our purpose. The following noise elimination operations are specialized according to the image quality, but all processes are done only on a SYM-picture.

In section IV-4, MOLD Theory is applied to the field of digital communication, and the redundancy reduction coding method on an LSC-picture is described. The information contained in an LSC-picture is less than that in the original bi-level image, so that it seems better for coding to work on an LSC-picture than on the original. A new concept as to error detection and correction is proposed in this coding method, and the compression ratio, the influences of the transmission errors, are experimentally investigated.

The information processing on a SYM-picture is, again, described in section IV-5, and is an example of the application of the theory to hand sketched line drawings. Considering the tiresome work of the input process for humans, there is a great need for machines to handle the hand sketched line drawings automatically. First, we eliminate the characters which have different features from the others. For the remaining diagrams, the feature points (which are crossing points, ending points and bending points) are extracted with their connectivity information. Then the low level descriptions of the diagrams consisting mainly of topological information are constructed, and a simple process is done on this description. Finally, the image is reconstructed. These processes are applied only to a SYM-picture, and the process would seem to be basically faster than any process based on original bi-level images.

Through these applications, we intend to show the effectiveness and usefulness of a SYM-picture or an LSC-picture, not only in the field of information processing, but also of digital communications. After the success of this type, the importance of the pattern generating constraints is, we believe, proved experimentally.

IV-2 Mesh Oriented Line Drawings Theory (MOLD Theory)

IV-2-1 Object World and Terminology

The world is assumed to be the figures of lines. In other words, it is the set of all the results which are the loci of pen movements. For example, our world includes characters, graphs and tables, although it excludes photographs with half-tone characteristics.

Our world has the feature that any pixels in the images have intrinsically two values, black, which represents the loci of the pen, and white, which is the remaining medium of the paper itself. Here it is worth commenting on dither images and pseudo-two valued images. They have intrinsically multi-level values or colors on the pixels, but they are conveniently expressed in two values.

First, we begin by considering a line. A line is taken to be the loci of the pen movement and it has two important features. One is the width of the line which is very closely related to the spot size of the pen used to generate it. The other is the length of the line which is related to the amount of the movement of the pen. Generally, the width of the line is almost constant as long as the pen used to generate the line remains the same, while the length varies according to the pen movement.

Even in the world of mathematics, there is no universal definition of lines. In geometry, the line is an undefined concept. Euclid said in his famous writings "Stoicheia" that the line is a length without any width and that the endpoints of the line are points, but this is only the concept of a line, not the definition of real world line patterns.

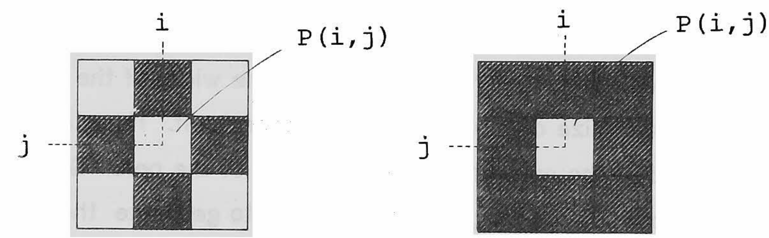
In the real world, a line does have width and a point does have some area although it is very small. The width of a line has a continuous value according to its coordinates and all the black points which are the parts of the line are connected. The world is an analog one.

On the other hand, in a digital world which is formed by sampling from the analog world, a point has a digital area and a line is associated with a digital valued width. The line has the smallest width, one pixel's width, which is the elementary unit in the digital world. Another feature of this world is the connectivity of the line. There are two different definitions as to the connectivity of a line, 4-neighbours and 8-neighbours.

Hereafter we limit the discussion only to the world which is handled by this theory. In the remainder of this section, we describe the terminology used, and in the next section the relation between the digital world and analog world is discussed in detail.

Strictly speaking, our world is assumed to be the set of lines in the digital world. The images are represented by M×N arrays, and each element (pixel) can take one of only two values, 0 or 1.

[Definition 4-1] 4-neighbours and 8-neighbours [16]
Let P(i, j) be a pixel of the given digital image at the coordinate (i, j). Then (i-1, j), (i, j-1), (i, j+1), and (i+1, j) are called the 4-neighbours of P(i, j) and denoted here E_p^4. In addition, (i-1, j-1), (i-1, j+1), (i+1, j-1), and (i+1, j+1) and 4-neighbours are called the 8-neighbours of P(i, j), and denoted E_p^8. Fig.4-1(a) and (b) show 4- and 8-neighbours, respectively. □



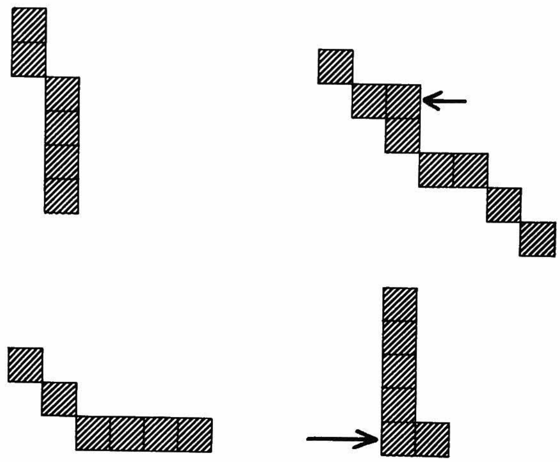
(a) E_p^4 4-neighbours (b) E_p^8 8-neighbours

Fig.4-1 4-neighbours and 8-neighbours.

[Definition 4-2] Simple arc [17]
For the sequence of the points (pixels) $A=\{P_1, P_2, ---, P_n\}$ such that,
(1) $1 \leq \forall\, r, s \leq n,\ P_r = P_s$ if and only if $r = s$
(2) $P_r \in E^8_{p_s}$ if and only if $r = s+1$ or $r = s-1$ □

Fig.4-2 shows examples and counter examples of simple arcs. In Fig.4-2(b), the pixel indicated by the arrow disturbed condition (2) of the above definition. A simple arc is considered to be the line with the narrowest width in the digital world.

Next, we define lines with any size of width.



(a) Examples of a simple arc

(b) Exapmles which are not simple arcs

Fig.4-2 Simple arcs.

[Definition 4-3] Region

The set of pixels $\mathbf{A}=\{P_1, P_2, \cdots, P_l\}$, such that

(1) for $1 \le \forall r, s \le l$ (r, s : integer) $P_r = P_s$ if and only if $r = s$

(2) for $1 < \forall s < l$, P_s, $\exists P_r, P_t$ $\quad 1 < r, t < l$
such that $P_r, P_t \in E^4_{P_s}$ and $P_r \in E^8_{P_t}$

(3) **A'** (**A**'s complement) must satisfy the above two conditions (1) and (2). □

Condition (1) assures that there does not exist the same pixel in the region, condition (2) defines the connectivity of the region and condition (3) gives the restriction that the region must be next to another region. In particular, if the region consists of black pixels, it is called "black region", and of white pixels, "white region".

Fig.4-3 shows examples of black regions and white one. In this figure, the black pixels' set at the bottom-left is a counter example because the pixel indicated by an arrow disturbed the above condition (2).

We can classify most images into two kinds of regions. One is represented as the background and the other is the lines. Indeed, this classification is independent of the

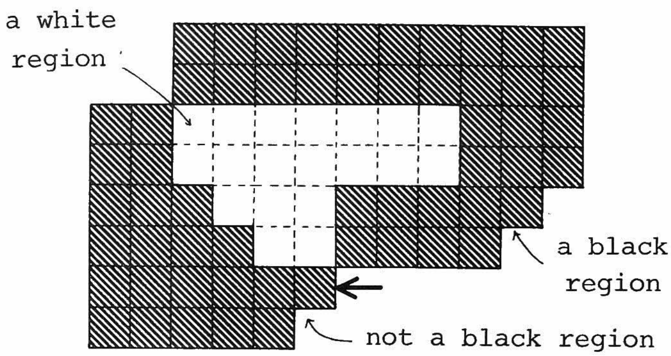


Fig.4-3 Regions.

color, black and white, of the regions, but in most cases, the color of the regions of the same kind is the same. After this, we call the latter class of the regions "digital lines" which means the lines in the digital world.

Examples of digital lines are shown in Fig.4-4. (a)-(c) are the examples of digital lines while (d)-(f) are not digital lines. (d) and (e) disturb the condition (2) in the above definition and (f) does (3). (a) and (c) are examples of the digital lines with the narrowest width. Digital lines are different from the simple arc since they can represent any lines with any width, so that the definition of the digital line is more general than that of the simple arc.

To prepare the definition of the width of the digital line, we must first define the observational unit.

[Definition 4-4] Unit mesh

As in Fig.4-5, the pixels in the digital image are placed in a square in size $m \times m$. The original digital image is then considered to be the new image with size $1/m \times 1/m$, and each $m \times m$ pixels' set is called a "unit mesh", and this image also called "mesh image", where m is greater than or equal to 1. □

[Definition 4-5] Window

A certain set of pixels with the same size and shape as the unit mesh which can be located in any place in a digital image is called "window". □

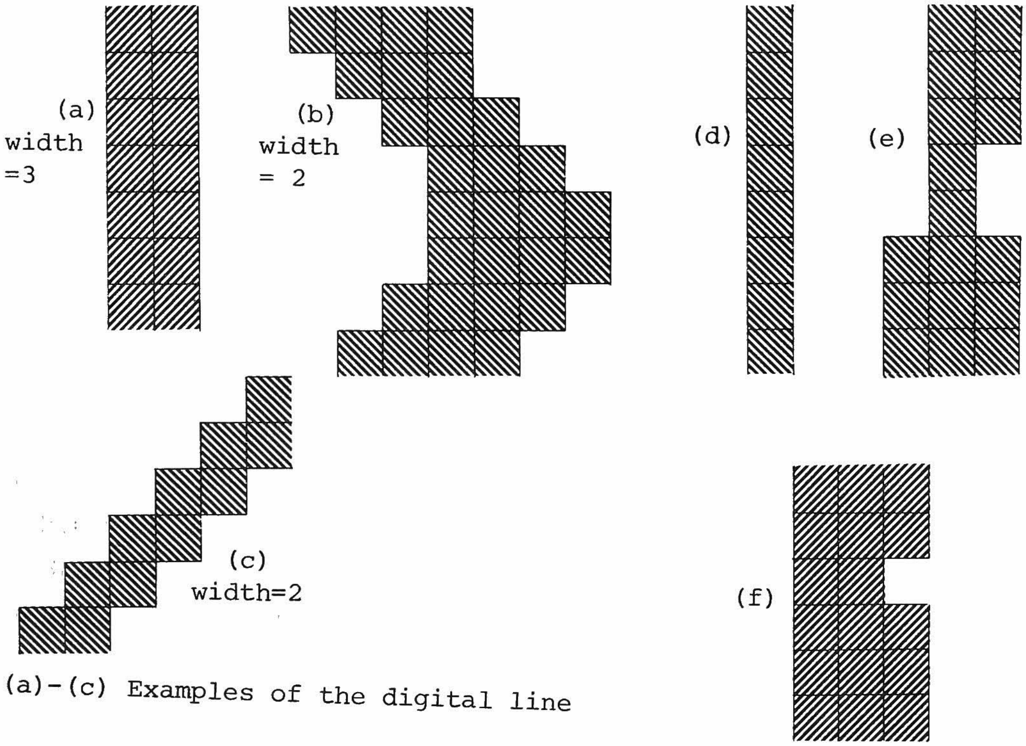


Fig.4-4 Digital lines.

The difference between the unit mesh and the window is that the former is fixed while the latter can be movable, pixel by pixel, in the digital image domain.

[Definition 4-6] The width of the digital line

Given the digital line, we can observe it in a window of a ceratin size. Moving the window to any position, it may be proved that there exist in the window at most one black region and one white region. Varying these windows' sizes, the maximum size of the window defines the width of the digital line. □

For the digital lines shown in Fig.4-4, the width is also denoted. The definition depends on the square window, so that the width would differ from intuition. This is because the width of the vertical and horizontal lines are $1/\sqrt{2}$ times as narrow as those of diagonal ones, but the original image is represented as the set of the square pixels. Therefore, our definition about the width of the digital line is more natural and effective than the other definitions depending on circles etc.

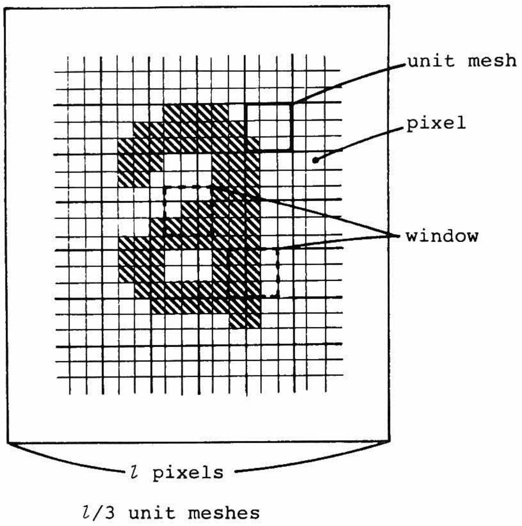


Fig.4-5 Mesh image.

At the end of this section, we must summarize the world to be handled in our theory. The world is assumed to be constructed from the set of digital lines, i.e. the world of line drawings. And also it is the digital world with only two values, black (1) and white (0).

Probably a question arises in your mind as to how we can actually generate this kind of world, or what constraints are necessary to transform the real analog world into the digital world. The next section answers these questions.

IV-2-2 Image Obtaining Constraints

IV-2-2-1 Constraints of the Sampling Process

The purpose of this section is to establish what constraints are necessary to generate the digital lines. Generally speaking, we obtain the digital image by sampling the analog image, so our purpose is also to determine the constraint of the sampling process.

First, the location of the digital lines in all of the line drawings in the digital world must be made clear. A well known theorem on the transformation between the analog world and the digital world is the sampling theorem originally developed for one dimensional wave forms. It assures that if the sampling intervals are finer than 1/2W (where W is the highest frequency in the waves), the digital waves transformed according to the sampling process are theoretically identical to the original analog ones. This theorem can be easily extended to the two dimensional case [18]. To explain the two dimensional sampling theorem intuitively, the sampling interval is fine enough not to miss the line with the narrowest line width. In other words, any line in the image has at least one sampling spot in the direction perpendicular to the line direction. The digital image generated in the sampling process satisfying this condition is proved to be identical to the original analog image.

Given arbitrary line drawings in the analog world, image sampling fine enough to satisfy the sampling theorem is easy. We call the line included in such a digital image "arc". This set of the arcs, the set of the digital lines and the set of simple arcs are denoted **U**, **D** and **A**, respectively.

Indeed, our theory treats U as the universal set. Otherwise, we cannot reconstruct the image exactly. It is clear that $\mathbf{U} \supset \mathbf{D}$ and $\mathbf{U} \supset \mathbf{A}$ and, from the definition in the previous section (IV-2-2), $\mathbf{A} \cap \mathbf{D} = \phi$ and $\mathbf{U} \supset (\mathbf{A} \cup \mathbf{D})$. These relations are illustrated in Fig.4-6.

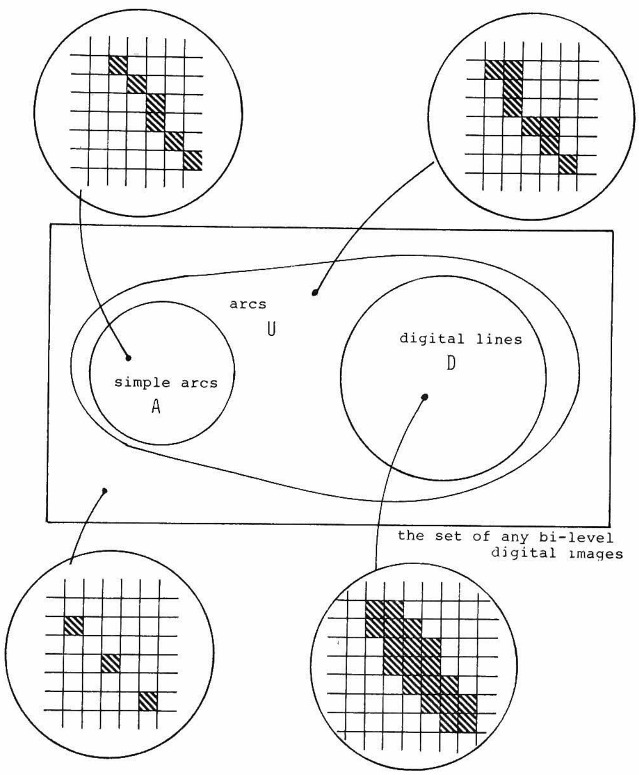


Fig.4-6 Relation among the sets of arcs, digital lines, and simple arcs.

This figure also shows a typical example of each set.

Our object world, the set of digital lines, is only a subset of the set **U** introduced by the two dimensional sampling theorem. To restrict the mapping from the analog world to the set of digital lines, a more powerful constraint than that of the sampling theorem is needed. The first theorem of this theory is concerned with this constraint.

[Theorem 4-1]

The line in the analog world is mapped to one element in the set of digital lines if and only if the sampling intervals are finer than, or equal to, twice of those specified by the sampling theorem.

(proof)

As stated, the sampling intervals specified by the sampling theorem give the constraint that the narrowest line in the image must have at least one sampling point on its width, whilst the restriction of this theorem is at least one half of these intervals. This means that the narrowest line has at least two sampling spots in its width.

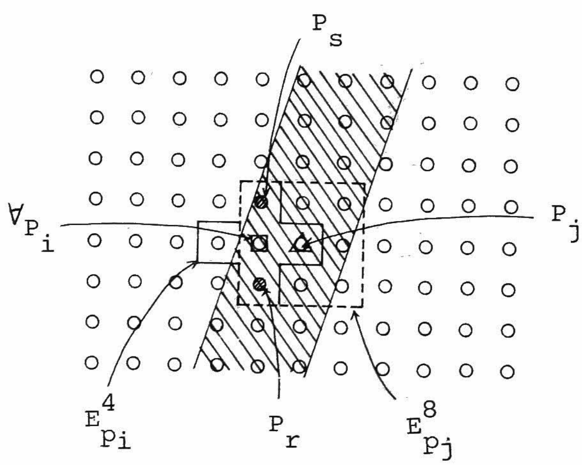


Fig.4-7 Constraint for the sampling process.

First, we show the necessity. Assume that the set of the sampling points on the line is **A**={P_i}. On the basis of the assumption that the sampling intervals satisfy the condition of this theorem, at least two sampling points are temporarily assumed, without any loss of generality, to exist on one horizontal direction as shown in Fig.4-7. One sampling point is selected arbitrarily and denoted P_i . For P_i , $\exists P_j$ in the same horizontal direction, such that $P_j \in E^4_{p_i}$.

Also $\exists\ P_r$, P_s such that P_r ,P_s $\neq P_j$, P_r , $P_s \in E^4_{p_i}$ and P_r ,$P_s \in E^8_{p_j}$. Now, P_r , P_s and P_i are in the same vertical direction and the constraint above is satisfied in the vertical direction, so that at least one of P_r or P_s is included in **A**. In the set **A**, $\forall\ P_i$, $\exists\ P_l$, $\exists\ P_m$ such that P_l , $P_m \in E^4_{p_i}$ and $P_l \in E^8_{p_m'}$, hence **A** is a digital line. The same discussion can be shown in the case of the set **A'**, the complement of the set **A**. Hence the constraint is a necessary condition.

On the contrary, if the digital line **A** is given, from the **Definition 4-3**, $\forall\ P_i \in$ **A**, $\exists\ P_j$, $P_k \in E^4_{p_i}$ and $P_k \in E^8_{p_j}$. This assures that P_i has at least one pixel in **A** in the horizontal direction and at least one in the vertical direction, which directly leads to this constraint. □

According to this theorem, given a line with any width, the sampling intervals can be selected depending on its width. In other words, the width of the line in the analog world is only measured in the dimension of length (mm) and its width in the digital world is measured with the quantum of a pixel. The key is the sampling resolution.

Facsimiles have become widespread as an input device of computers. Generally, the facsimiles cannot change the resolution arbitrarily, so that this relative relation becomes the constraint on the width of the line drawn in the real world. If we feed the image through the facsimile in the fine detail mode (about 7.7 lines/mm) recommended by CCITT SGXIV, the lines with width more than 0.25 mm coincide with the digital lines. This example indicates that the constraint laid by **Theorem 4-1** is not so far from practical use.

Fig.4-8 are the test charts recommended by CCITT for facsimile transmission. For these charts, all the lines are sampled into digital lines except those in NO.7 (Japanese document) by the facsimile's fine detail mode. In NO.7 the Chinese characters of Mincho type have very thin horizontal lines and they are disturbed the constraint of the **Theorem 4-1**, but if we feed it via a device of about 13 lines/mm resolution, then the constraint is satisfied and the resultant lines can be digital lines. These experiments show us that the constraint is not so strong a one.

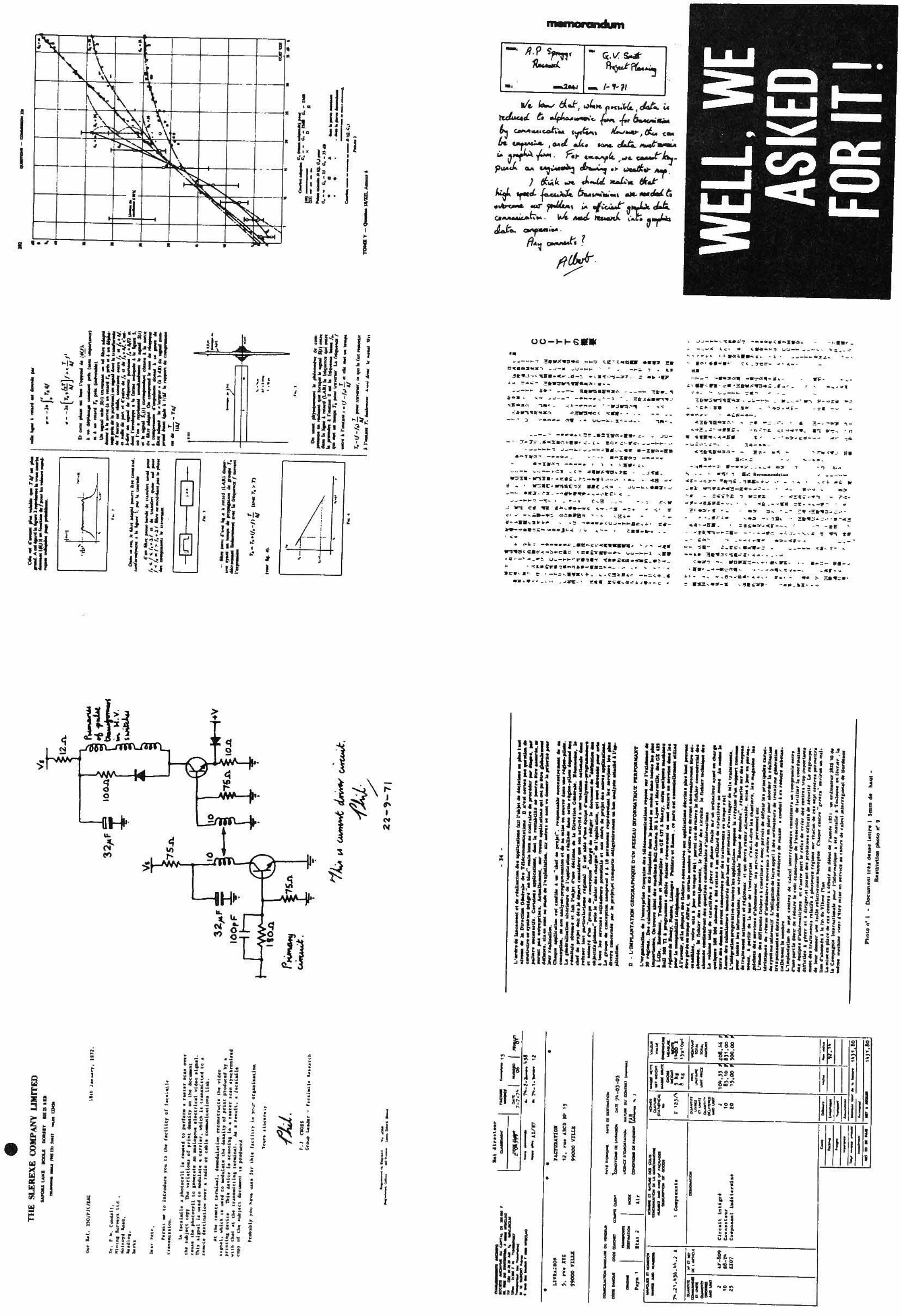


Fig.4-8 CCITT's test charts.

IV-2-2-2 Assumption about Measuring Meshes and the Line Width

When we draw the lines, we use a pencil or a pen. The tip of these tools has an area which will hereafter be referred to as the "penpoint". The width of the lines is very much related to the size of the penpoint and is a very important feature in processing the lines or line drawings.

We believe that there exists a constraint due to the tool which generates the patterns on the higher level of the physical constraints. More generally, we must take advantage of this constraint in the field of information processing and of digital communication. Such constraints are called "pattern generating constraints" or "information boundary conditions" as already described in section II-3.

In our line drawings' world, this constraint appears as the width of the line. To utilize this phenomenon, the observation unit mesh must be selected according to the width of the line. To construct the theory, there must be an assumption between the size of the observation unit (actually the unit mesh) and the width of the digital lines.

Various assumptions will be able to use this constraint, but we must take the actual processes in computers into consideration. For MOLD Theory, we select an assumption called the **"Fundamental Assumption"**.

[Fundamental Assumption]
The width of the digital line (n) must be greater than or equal to the size of the unit mesh (m). □

Hereafter, our theory is based on this **Fundamental Assumption**. In the next section, we define legal patterns, the most important elements in MOLD Theory.

IV-2-3 Legal Patterns

We assume two things in the following discussion.

(i) The sampling process satisfies the constraint presented in the **Theorem 4-1.**

(ii) For the size of the unit mesh and the width of the digital line, the **Fundamental Assumption** will be satisfied, and the narrow gaps (complement of the lines) among the lines should also be treated as digital lines.

In these conditions, we can define the legal patterns in the unit mesh.

The first thing we present here is the relative relation of the positions between the digital lines and the unit mesh.

[Theorem 4-2]
In the case that the conditions (i) and (ii) are established, the positioning relations between the digital lines and the window are restricted to the following three cases (examples are shown in Fig.4-9).

(1) The digital line fills the window completely (Fig.4-9(a)).
(2) The digital line partly fills the window (Fig.4-9(b)).
(3) The digital line and the window are disjointed (Fig.4-9(c)).

(proof)
From the **Definition 4-6** and condition (ii), the cases are prohibited where the window includes the digital line completely (Fig.4-9(d)) and where the window contains two or more digital lines as shown in Fig4-9(e). These five cases are exhaustive, hence the theorem is proved. □

[Definition 4-7] Legal patterns and illegal patterns
The pattern set logically occurring in the unit mesh is denoted **S**. Consider the logical function **K**, which is the mapping from **S** to $\{0,1\}$, as follows.
For $\forall\, x \in S$

$$K(x) = \begin{cases} 1 : \text{in the case } x \text{ matches the constraint} \\ 0 : \text{in the case } x \text{ disturbs the constraint} \end{cases}$$

This function **K** is called the "constraint function". The pattern set which consists of only

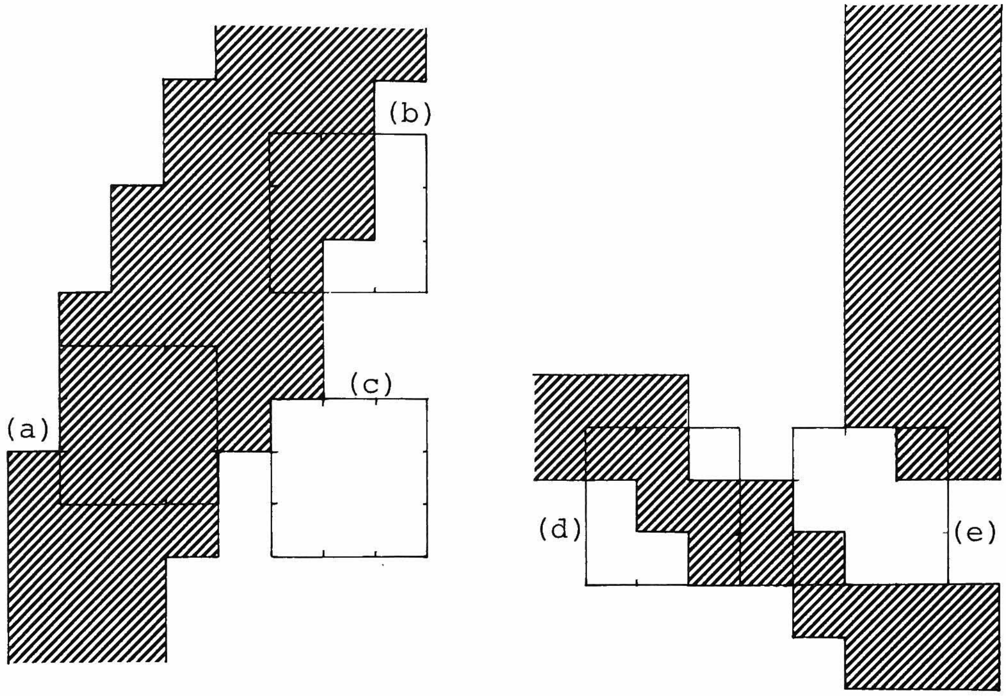


Fig.4-9 Relations between the window and digital lines.

the patterns with "K equal to 1" is called "legal pattern set (U)" and each element of it is called "legal pattern", and the pattern set which has "K equal to 0" is called "illegal pattern set (V)" and called "illegal pattern". The following relations are clear.

$$S = U \cup V, \quad U \cap V = \phi$$

To estimate the power of the constraint, we also define the "constraint ratio" as follows.

$$\eta = \frac{\text{the number of the elements in V}}{\text{the number of the elements in S}} \qquad \square$$

Theorem 4-2 is an implicit description of K, that is, the constraint due to the digital line. Legal patterns consist of the following three kinds.

[Legal patterns for the digital line]

(1) All the pixels in the unit mesh are black (the case (1) in the **Theorem 4-2**).

(2) One black region and one white region, both are adjacent to the edges of the unit mesh (the case (2) in the **Theorem 4-2**)

(3) All the pixels in the unit mesh are white (according to the case (3) in the **Theorem 4-2**) □

Now, we can count the number of elements in the legal pattern set **U**. The next theorem shows how to do it.

[Theorem 4-3]

The problem as to counting the number of legal patterns under the conditions (i) and (ii) is transformed into the following problem in natural numbers.

"Given an integer k , find the integer sequences b(i) such that

$0 \leq b(i) \leq m\ ,\ \ i > j \ \ \rightarrow \ \ b(i) \leq b(j)$

where $1 \leq i, j \leq m\ ,\ \ 1 \leq k \leq \lfloor m^2/2 \rfloor$

b(i) : integer

m : the size of the unit mesh

$\lfloor x \rfloor$: the largest integer which is not greater than x

(proof)

The positional relations (1) and (3) in the **Theorem 4-2** have generated one legal pattern, respectively, so that case (2) is the problem.

The unit mesh is square and the image is bi-level, so that there are two symmetries, one is black and white, and the other is a rotation of 90°. We assume without loss of generality that the left-upper pixel of the unit mesh is black and that the number of black pixels is decreasing as the row position of the unit mesh goes down (in other words, increasing the row number as shown in Fig.4-10). If we denote the number of black pixels b(i) (i=1,2,---,m) (where i represents the row number shown in Fig.4-10), then b(i) satisfies the condition

$0 \leq b(i) \leq m$ and $i > j\ , \rightarrow b(i) \leq b(j)$

The symmetry in 'black and white' brings us the condition that $1 \leq k \leq \lfloor m^2/2 \rfloor$ Because m^2 represents the number of all pixels in the unit mesh, the number of the black pixels which is delivered to each row is at most $\lfloor m^2/2 \rfloor$. □

The problem presented in the **Theorem 4-3** is easy to solve. If we specify the maximum number, i.e. $b(1)=l$, $1 \le l \le m$, the problem may be transformed to the same problem with smaller k.

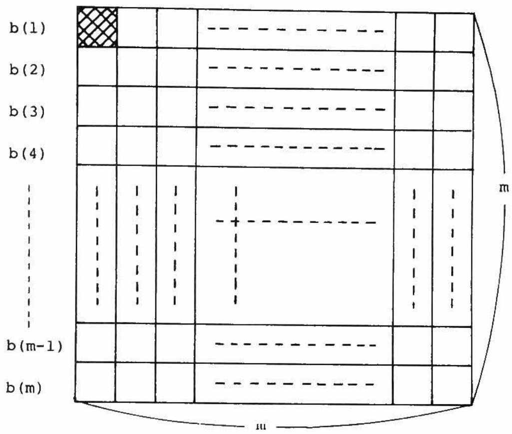


Fig.4-10 Relation between b(i) and the unit mesh.

"Given a integer $k - l$ (≥ 1), find the sequences b(i) such that

$0 \le b(i) \le l$, $i > j \rightarrow b(i) \le b(j)$

where $2 \le i, j \le m$ "

By using a recursive procedure, we can decompose the given problem into a problem which may be easily solved.

Generally we find a number of sequences $b_h(i)$, h = 1, 2, --- . To each sequence the legal pattern with a black pixel in its left-upper position corresponds. Usually, each legal pattern has eight symmetrical legal ones respectively. They are the symmetry of 90° rotation and of 'black and white'. There exist two exceptions.

(1) The sequence $b(i) = k$ for all i and $b(j) = m$, $j = 1$, --, k and $b(i) = 0$ $i = k$, --, m $k < \lceil m/2 \rceil$ are identical because they have a symmetry of 90° rotation (See Fig.4-11(a)(b)).

(2) If m is even, the sequence $b(i) = m$ for $1 \le i \le m/2$ and $b(i) = 0$ for the other cases as shown in Fig.4-11(c). This pattern does not have a symmetry pattern as to the black and white.

If the number of the sequence b(i) is h, then the number of the legal patterns N_u are obtained by the following equation.

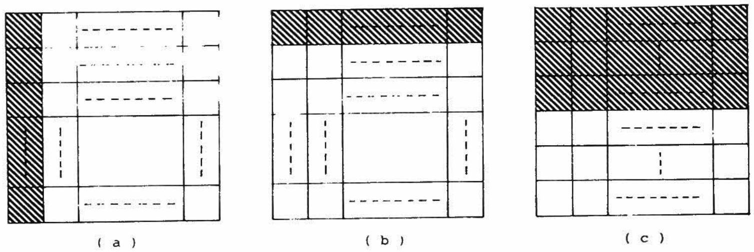


Fig.4-11 The case that b(i) does not correspond to the legal pattern.

$$N_u = \begin{cases} 8 \times (h-(m-1)/2) + 2 & m : \text{odd} \\ 8 \times (h-m/2) - 2 & m : \text{even} \end{cases}$$

Table 4-1 represents the results for counting the number of the legal patterns and of b(i) sequences for various m. It also contains the number of the elements in S and the constraint ratio.

Table 4-1 shows that as the mesh size becomes larger, the constraint ratio rapidly increases and almost attains the saturation point i.e. 1.0 for m=4. But under the condition

Table 4-1 Number of legal patterns with various values of m (size of the unit mesh).

m	number of element in S	number of sequence $b_h(i)$	number of legal patterns	constraint ratio	cost	constraint rato / cost
1	2^1	1	2	0.00	1	0.00
2	2^4	3	14	12.50	4	3.13
3	2^9	9	66	87.11	9	9.68
4	2^{16}	38	286	99.56	16	6.22
5	2^{25}	125	986	99.99	25	4.00
6	2^{36}	490	3894	99.99	36	2.78

that the **Fandamental Assumption** is established, the size of the unit mesh is the lowest bound of the digital line. Therefore the constraint on the sampling process grows stronger or the lines drawn must be wider. Much worse, the amount of information of the digital image is larger due to the fine sampling resolution. There is much disadvantage in making the size of the unit mesh larger.

Now, the problem is what size of unit mesh is optimum. One measure in favor of making the size of the unit mesh larger is the constraint ratio. As the other measure to limit it, we select a cost proportional to the amount of information. We assume that the unit cost is the cost of the amount of information to be sampled in the intervals specified by the sampling theorem, and that if we make this interval 1/h, the cost is h^2 times as much. Then we calculate a kind of cost-performance i.e. the former is divided by the latter.

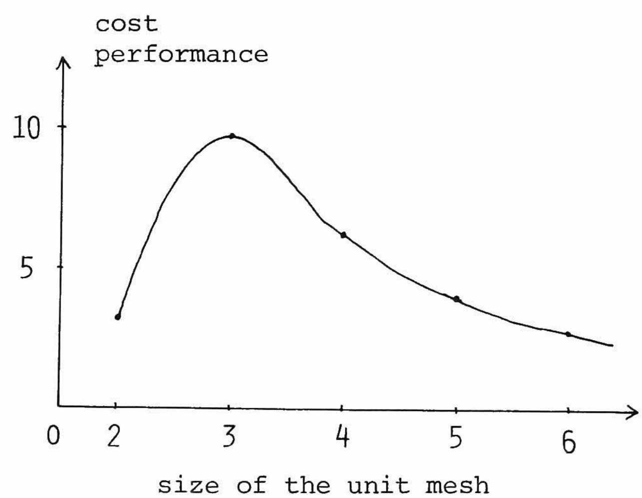


Fig.4-12 Graph of cost-performance vs. size of the unit mesh.

Fig.4-12 is the resultant graph for this parameters versus the size of the unit mesh. From this figure, we see that m=3 is optimum.

From now on, the size of the unit mesh is fixed at 3. Consequently, the width of the digital lines must be greater than 3. If we use the facsimile in the fine detail mode (about 7.7 lines/mm), the lines must be drawn with the penpoint of size about 0.4 mm. For a propelling pencil, its penpoint size is usually 0.5 mm, so this constraint is natural in the real world. Fig.4-13 shows the legal pattern set with all its 66 elements. The number with each pattern is the order of its population in the statistics of the CCITT NO.1 (English

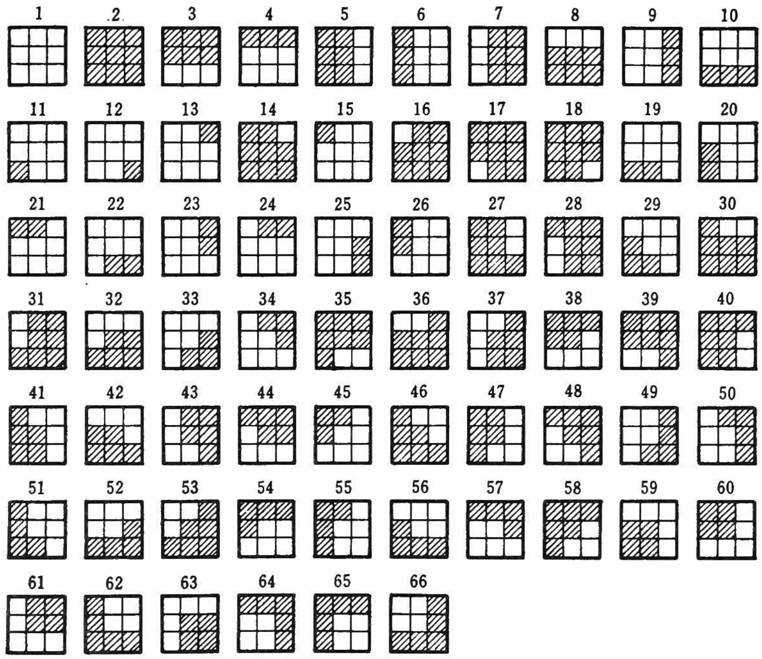
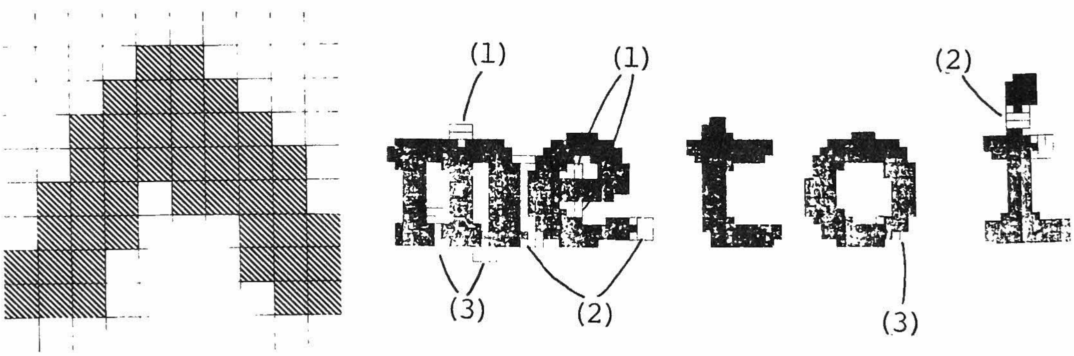


Fig.4-13 Legal patterns (m=3).



(a) tip point

(b)

Fig.4-14 Examples where illegal patterns actually occur.

letter) shown in Fig.4-8. The first 66 are exactly identical to our legal patterns. The legal patterns for m=2 and the sequence representing the legal patterns for m=4, 5 are shown in Appendix-B.

In the field of digital geometry, there is a definition of "connection number" [20,21,22]. The legal patterns without 1 and 2 in Fig.4-13 have the characteristics that the center cells of them have the connection number 1. This means that the color of that cell can be changed without affecting the whole connectivity. The set of patterns with connection number 1 includes the set of legal patterns bar two, so that the latter has more constraints than the former does.

If the image contains only the digital lines not crossing each other, all the patterns occurring in the unit mesh must be legal patterns. But, generally, there are digital lines crossing each other. In this case, there is a possibility for illegal patterns to occur. They are located especially around the crossing points of the digital lines. An example is presented in Fig.4-14(a). This is an intrinsic problem for our theory, but the number of illegal patterns which actually occur is very small, therefore there is no problem in processing the digital images.

IV-2-4 Transcription of Real World Image into Symbols (SYM-picture)

The legal patterns introduced in the previous section are theoretically derived from the constraint of the sampling process and the **Fundamental Assumption.** If these conditions are established, almost all the patterns in the unit meshes may be legal ones. We investigate this in the real world.

The image data for experiments are CCITT's test charts shown in Fig.4-8 and digitized by a digital facsimile with resolution of about 7.7 lines/mm. The input images have a size of 1,728 × 2,352 pixels, and the statistics of the patterns in the window (the size is 3 × 3) are obtained by moving it one pixel by one as in the scanning of TV. The total number of each statistics is (1728-2) × (2352-2) = 4,056,100, and there are $2^9 = 512$ patterns in the window which may occur logically. Table 4-2 shows the number and the ratio occupied by the legal patterns.

All the images except NO.7 (Japanese document) satisfy the previous conditions and

Table 4-2 Cumulative ratio of legal patterns.

	ratio of legal patterns	without W pattern	resolution (lines/mm)
CCITT NO.1	99.83	96.82	8
CCITT NO.2	99.96	99.24	8
CCITT NO.3	99.72	96.76	8
CCITT NO.4	99.09	95.04	8
CCITT NO.5	99.57	95.19	8
CCITT NO.6	99.77	96.15	8
CCITT NO.7	97.60	83.54	8
CCITT NO.8	99.89	99.76	8
CCITT NO.7	99.46	96.74	13

the results assure that the legal patterns are well fitted to the real world patterns. For the image of NO.7, the horizontal lines included are very thin and the lines cannot become digital lines through the input process of the digital facsimile. Thus the ratio of the legal patterns is relatively low, but if we sample this image in about 13 lines/mm, the conditions of the theory are satisfied and the ratio of the legal patterns is as high as those of the others.

Nevertheless illegal patterns do occur. Fig.4-14(b) are examples where illegal patterns are in the image. The illegal patterns are classified into the following three categories according to their causes.

(1) The tip points where two or more digital lines cross in an acute angle with each other.

(2) The illegal patterns which occur from the fraction of the sampling spot in the input facsimiles.

(3) The local place where the **Fundamental Assumption** is disturbed.

The categories (2) and (3) are the results of some image distortions due to the mechanical and physical process. It is desirable for good image quality to eliminate these kinds of illegal patterns, but the illegal patterns of the first category (1) cannot be eliminated without causing the image some distortions. Fig.4-15 is examples of the case

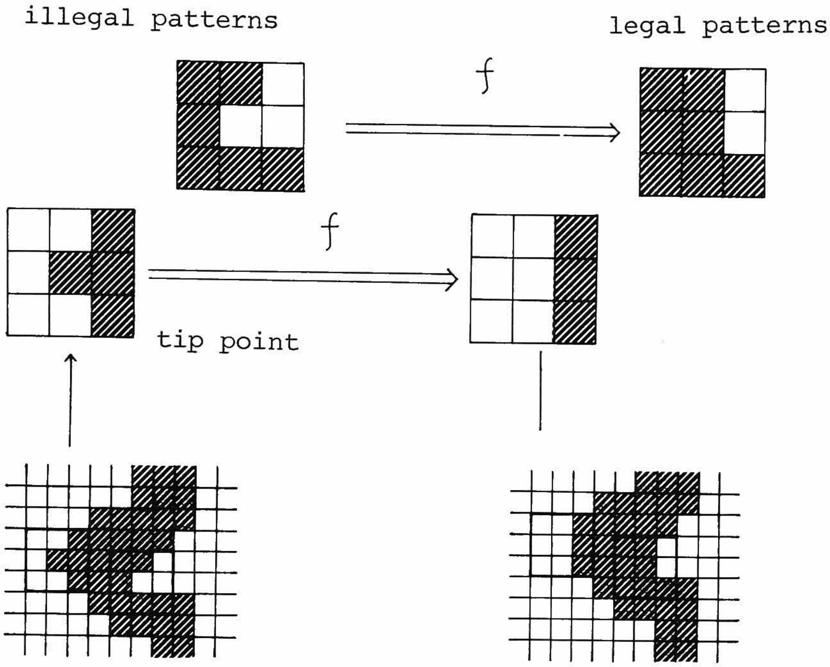


Fig.4-15 Elimination of illegal patterns in category (1).

that the illegal patterns in the category (1) are replaced by one of the legal patterns. The tip points or acute angles are seen to be a little rounded. We do not strictly mind these distortions, because the width of the digital line is more than three, and the number of illegal patterns is very small compared with the whole number.

Considering the discussion above, we introduce a new representation for the bi-level digital image of the line drawings.

[Definition 4-8] Picture Represented by Symbols
The digital image is divided by a mesh whose position is fixed, and the patterns occurring in the unit meshes are restricted to be legal patterns and we call them "symbols".

We regard each unit mesh as a PIXEL, so that the resultant picture[1] expressed by PIXELs has the size of one third for each direction and the value of the PIXEL is assigned

1) "Image" is used in real world and "picture" is in the model world

to be the ordinal number of the legal pattern. This picture is called "picture represented by symbols" or abbreviated "SYM-picture". Corresponding bi-level images are denoted $(\text{SYM-picture})_2$ if there is some necessity to distinguish them explicitly. □

When we transcribe a SYM-picture from the digital image, there occurs the problem of how to handle the illegal patterns. In other words, we must select one of the legal patterns most suitable for each illegal pattern. Formally, we can define the transcription function as follows.

[Definition 4-9] Transcription function

For $\forall s \in V$ (the set of the illegal patterns), $\exists f$ such that

$f(s) = p \quad p \in U$ (the set of the legal patterns)

f is called a transcription function. □

We argue that almost all the illegal patterns are evidence of distortions of the image. In this process, unless the replacements specified by the transcription function are good enough for the resultant quality, the discussion will not be supported. In this view, the transcription function has great influence on the image quality.

An example of the transcription function is as follows.

(An example of the transcription function)

The black and white pattern in the unit mesh is considered to be a binary code word and the distance between the two patterns in the unit meshes is defined by the Hamming distance between the corresponding two code words (See Fig.4-16). Then

for $\forall s \in V$, we select $p \in U$ such that

the Hamming distance between s and p is minimum.

If there are more than two patterns for p, we select the legal pattern with highest population in the statistics of the image. □

This simple example of the transcription function is relatively good for a high quality image, but if the image quality is not so good, the result of this function is a little worse than the original. This process intrinsically needs information about the image qualities, and we discuss this problem in more detail in section IV-3.

Finally, a $(\text{SYM-picture})_2$ with this transcription function is shown in Fig.4-17 with its original bi-level image. We cannot find any difference between these two images at a

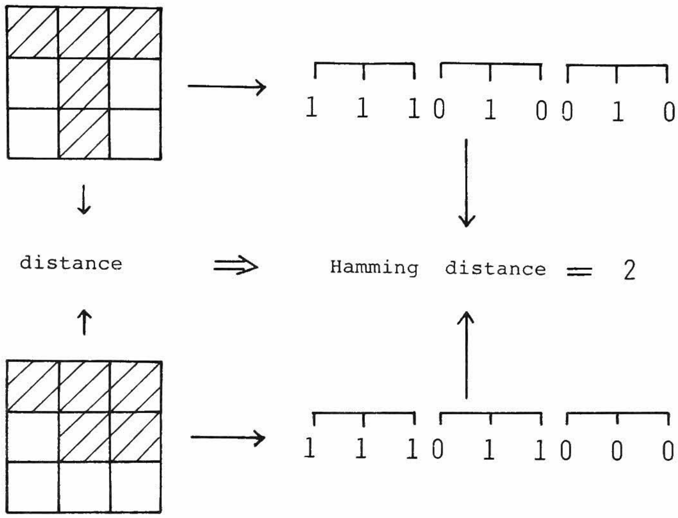


Fig.4-16 Distance between two patterns in unit meshes.

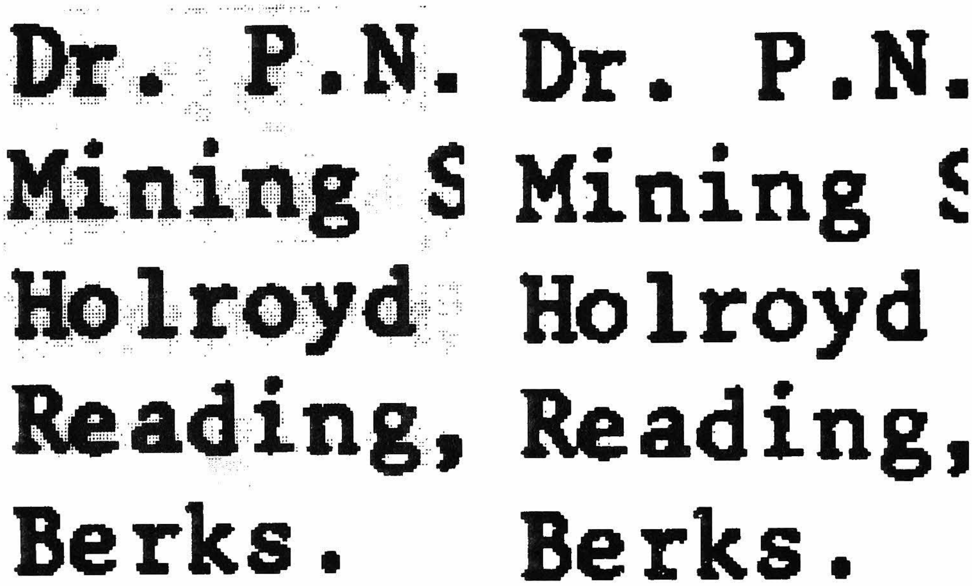
glance. Looking more closely, the small noises are eliminated, so that this process is not an approximation of the original image but an improvement of it.

IV-2-5 Picture Represented by Legal Symbol Connection (LSC-picture)

A SYM-picture may still have illegal patterns in the window just located between the unit meshes, because the mesh division is fixed and only patterns in the unit mesh are legal ones. In this section, we consider this dependency of the position of the mesh devision and eliminate all the illegal patterns in the image.

[Definition 4-10] Picture represented by legal symbol connections
Among all SYM-pictures, when the patterns in the window at any pixel position for their corresponding $(\text{SYM-picture})_2$ are all legal ones, these SYM-pictures are called "picture represented by legal symbol connections" or abbreviated "LSC-picture". The corresponding bi-level image is denoted "$(\text{LSC-picture})_2$ ". □

There are two ways to make LSC-pictures, as illustrated in Fig.4-18. One way is to make SYM-pictures first from the original bi-level images and then by introducing the



(a) original (b) (SYM-picture)_2

Fig.4-17 Example of a SYM-picture.

connection rules among symbols to transform SYM-pictures to LSC-pictures. This is called "fixed mesh method" and is described in sections IV-2-5-1 and IV-2-5-2.

The other method is to make the (LSC-picture)_2 directly from the original bi-level image. This method is called "scanning unit mesh window method" or abbriviated "scanning mesh method", and the details are described in section IV-2-5-3.

IV-2-5-1 Connection Rules of Symbols

In the process used to make SYM-pictures, illegal patterns in the window are occasionally eliminated because they are in the unit meshes. The other illegal patterns remain because they are not in the unit meshes but between the unit meshes. The constraint of "digital lines" is still available for such illegal patterns, so that there will exist some rules as to the connections of adjacent symbols.

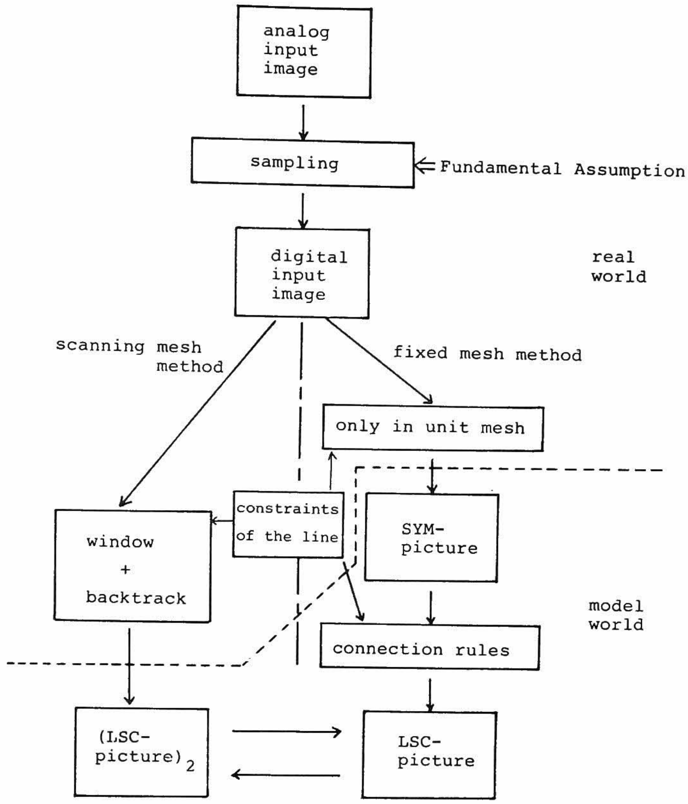


Fig.4-18 Schema of making LSC-pictures.

Before describing connection rules, some preparations are needed.

[Definition 4-11] Peripheral values

For the unit mesh, the outer-most pixels in either the horizontal or vertical direction are called "connection pixels". These connection pixels have peripheral values

(1) the kind of the pixel --- W (white) or B (black)

(2) the number of the run-length in the direction of the connection □

Fig.4-19 shows examples of peripheral values. (a) is a legal pattern and, (b) and (c) are the peripheral values for the vertical and horizontal direction, respectively.

The unit mesh is in the form of a square, so that we must consider the connection rules for at least four symbols displayed in Fig.4-20. For the region of these four symbols, there are sixteen different positions for the window. In a SYM-picture, only 4 windows are

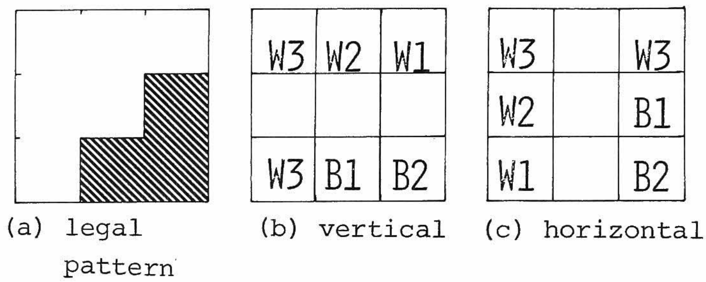


Fig.4-19 Peripheral values.

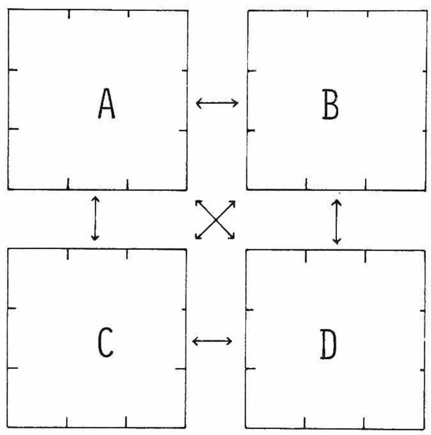


Fig.4-20 Four symbols for the connection rules.

legal patterns. To the contrary, all sixteen windows are legal patterns in LSC-pictures, i.e. there cannot exist any illegal pattern.

[Connection rules for symbols]

(i) The rules for the corresponding two pixels

1) In the case that the kinds of two pixels are the same

P1 : The acceptance condition is

the sum of the peripheral values ≥ 2

2) In the case that the kinds of corresponding two pixels are different

P2 : The acceptance condition is

each peripheral value ≥ 2

(ii) The rules for corresponding symbol pair

These rules are based on the rules for the pixels, that is, in order to apply these rules, the corresponding pixels must satisfy either P1 or P2 condition. The objective symbol pairs are A and B to C and D for the vertical direction, and A and C to B and D for the horizontal (see Fig.4-20 again).

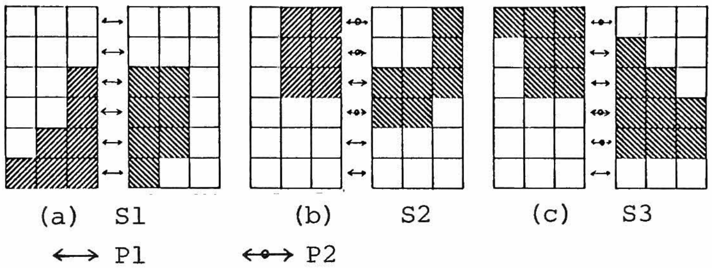


Fig.4-21 Connection rules.

1) In the case that all the connections for the pixels are P1s

  S1 : all this case are accepted (an example is shown in Fig.4-21(a))

2) In the case that the pixel connections include more than two P2s

  S2 : all are accepted except the case that two P2s exist at most two pixels' distance apart and the kinds of the pixels on the same side are not the same (see Fig.4-21(b)).

3) In the case the pixel connections include at least one P2

  S3 : all are accepted except the following two cases

   i) the peripheral value B1 is at most two pixels' distance from the P2 with B3 on the same side (see Fig.4-21(c))

   ii) the same case as i) with white pixels □

Our purpose is to make LSC-pictures, so that the question arises whether this connection rules are sufficient or not. The answer is the following theorem.

[Theorem 4-4]

SYM-pictures become LSC-pictures if and only if the connection rules above are satisfied at any position in the horizontal and vertical direction.

(proof)

nessecity:

We consider two symbols only in the horizontal direction because of symmetry.

1) In the case S1 of the connection rules

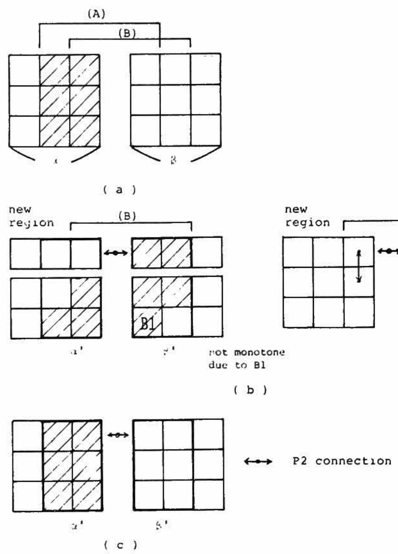


Fig.4-22 Explanation of the proof for Theorem 4-4 (necessity).

As shown in Fig.4-22(a), there are two windows (A) and (B) in the horizontal direction spanning two unit meshes. The pixel connections are all P1 connections, hence in the windows (A) and (B), the number of regions is at most the same in the unit mesh α and β, respectively. Each pattern in the unit mesh is legal, hence the patterns in those window are also legal.

2) In the case only one P2 connection exists (S3 of the connection rules)

It is easy to see that a P2 connection makes a new region usually consisting of only one pixel in those windows spanning two unit meshes. If this new region has the same kind of pixels in its 4-neighbours, the patterns in the window are legal. If the intersections between those windows and the unit meshes (α' and β') consist of only one kind of pixel, the new region does not make the legal pattern into an illegal one. S3 rule prohibits the occurrence of a pixel with peripheral values B1(W1) on the side of the black(white) pixel of the P2 connection which is at most two pixels' distance apart. This assures that the second condition, i.e. the intersection is monotone, is always established (see Fig.4-22(b)).

3) In the case that more than two P2 connections exist (S2 connection)

In this case, both S1 and S2 rules are applied. The P1 and P2 relation is discussed above, so that the relation must be investigated between two P2 connections. The connection rule S3 prohibits the two P2 connections having different kind of pixels on the same side. So that, as shown in Fig.4-22(c), the intersections between those windows and the unit meshes (α' and β') must be monotone, hence the patterns are legal patterns.

These 1), 2) and 3) are exhaustive. By taking advantage of the symmetry about the direction, if the connection rules are established for four symbols in the positions shown in Fig.4-20, it is proved that there cannot exist a window with an illegal pattern in it. Hence if the connection rules are satisfied at any position, there are no illegal patterns at all. Therefore, such a SYM-picture is an LSC-picture.

sufficiency:

We assume that an LSC-picture does not establish the connection rules, S1, S2 and S3, and show that this leads to contradictions.

1) P2 condition is disturbed

If the P2 condition is disturbed, an illegal pattern as shown in Fig.4-23(a) must occur.

2) S1 condition is disturbed

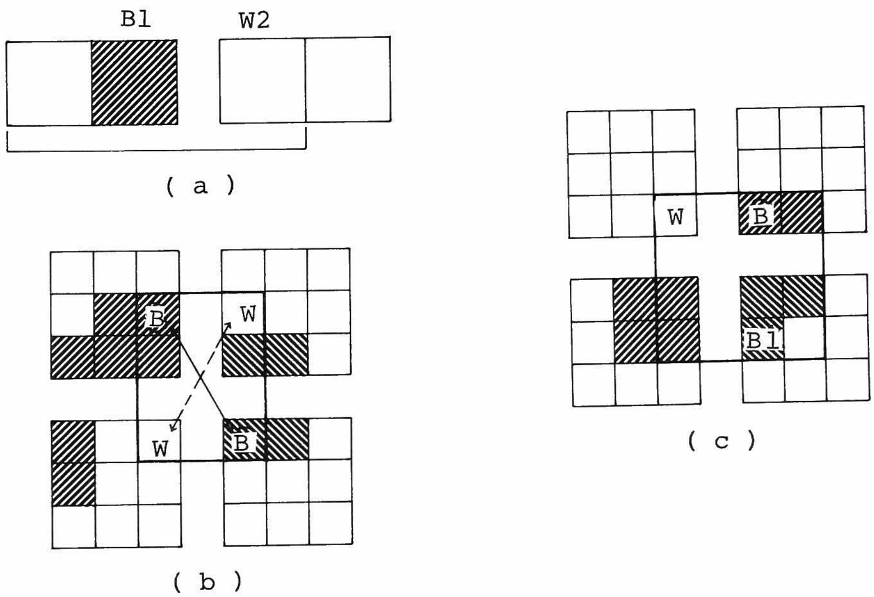


Fig.4-23 Explanation of the proof for Theorem 4-4 (sufficiency).

The P1 condition is always satisfied, and S1 case has no condition, so that this case can never occur.

3) S2 or S3 condition is diturbed

As shown in Fig.4-23(b)(c), there exist the illegal patterns. Hence a contradiction.

These 1), 2) and 3) are exhaustive. Hence it is proved that a LSC-picture must satisfy the connection rules. □

This theorem assures that we can make LSC-pictures by checking the symbol connections. Before describing this method, the combinations of symbols must be investigated in detail.

The number of adjacent two symbols which agree with the connection rules (we call it "legal connection") is 1,582 (this occupies 36.3% of all possible combinations 66^2 = 4,356), and of four symbols in the positions shown in Fig.4-20 is 384,562 which occupies 2.03% of all 66^4 = 18,974,736. These facts show us that the constraints of digital lines are also useful among the symbols. Then we investigate how to fit the connection rules to the real world. For the CCITT's test charts, the number of legal connections and their ratios are listed in Table 4-3. The ratios are almost all more than 95% except NO.7 which does not satisfy the **Fundamental Assumption**. These facts convince us that our MOLD theory fits the real world very well.

IV-2-5-2 Fixed Mesh Method

In the previous section, we defined the connection rules for symbols and proved that if a SYM-picture satisfies them at any position, it is an LSC-picture. In this section, the algorithm called "fixed mesh method" based on the above mentioned rules is presented. Given a SYM-picture, one of the symbols in it has the 'four relations' objective for the connection rules as is shown in Fig.4-24. If these relations do not satisfy the connection rules somewhere, the symbols due to these disturbances must be replaced by some other symbols which satisfy the connection rules. The most important problem is, then, that we cannot uniquely decide the symbols to be replaced. For example, in Fig.4-24 for the case that the relations (a) and (c) disturb the rules at the same time, the two symbols which are denoted A and B are the candidates to be replaced, but we do not have more information to decide which symbol to select.

Table 4-3 Cumulative ratio of legal connections.

	vertical direction		horizontal direction	
	ratio of legal connection	without W-W connection	ratio of legal connection	without W-W connection
CCITT NO.1	99.62	93.79	99.75	95.88
CCITT NO.2	99.97	99.42	99.96	99.31
CCITT NO.3	99.71	97.49	99.65	96.85
CCITT NO.4	98.58	94.01	99.16	96.41
CCITT NO.5	99.40	94.80	99.53	95.91
CCITT NO.6	99.75	96.72	99.78	97.03
CCITT NO.7	96.50	80.97	97.87	88.61
CCITT NO.8	99.86	99.69	99.92	99.83

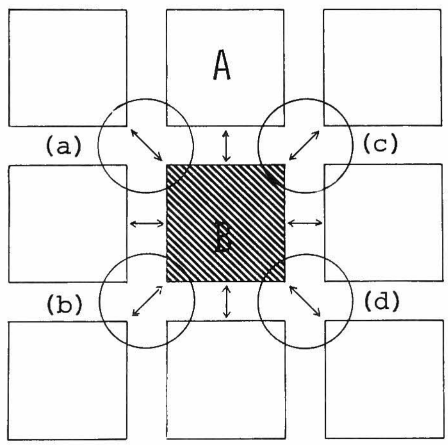


Fig.4-24 Four relations for one symbol.

One of the widely used technique to transmit or process the image is scanning. We take advantage of this scanning technique to give one solution to this ambiguity. This method called "fixed mesh method" because the objective picture is a SYM-picture which is produced in the condition of a fixed mesh.

[Fixed mesh method]

A SYM-picture is scanned one by one on the symbol. At the same time the processes below are applied.

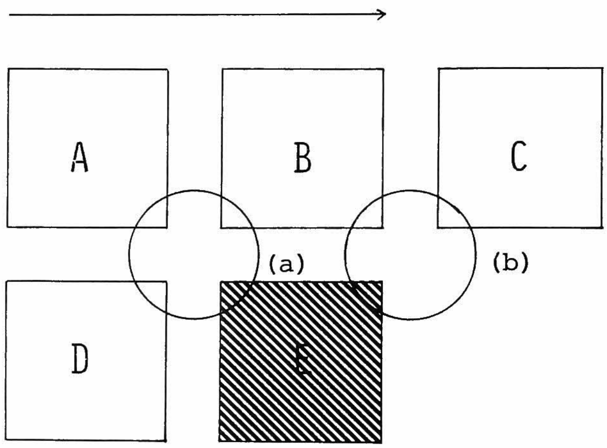


Fig.4-25 Fixed mesh method.

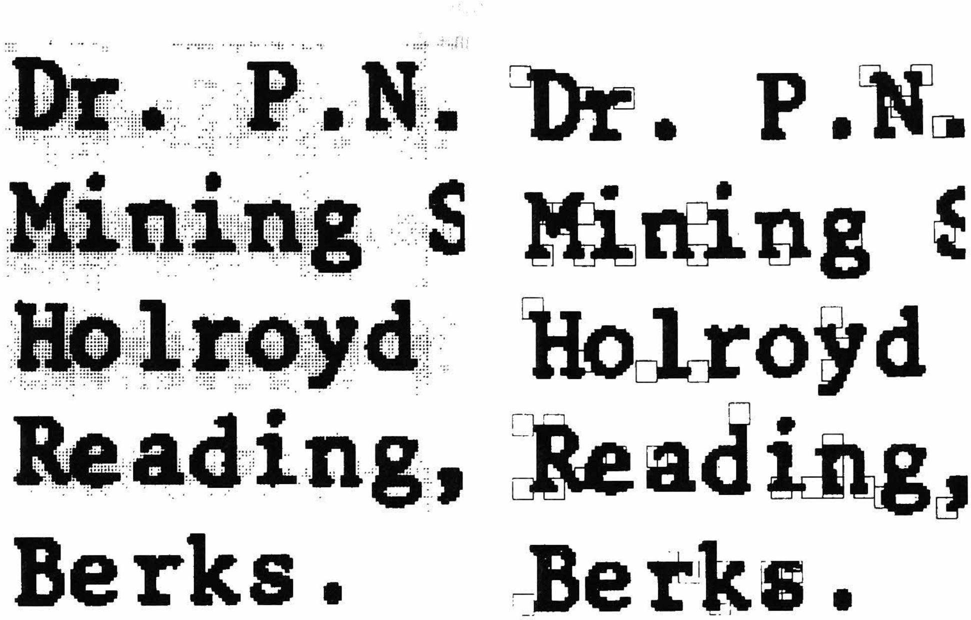
(i) For the symbols in the states of Fig.4-25 with two relations (a) and (b), it is checked whether they satisfy the connection rules or not. If they do, the process is finished and it proceeds to the next symbol specified by the scanning.

(ii) If there occur some disturbances in at least one relation, then we can correct them by replacing the symbol E. The reasons are as follows : the connections A and B, A and D, B and C, B and D have already been checked before, so that if the relations (a) and (b) disturbes the rules, it is due to the symbol E i.e. the connections A and E, B and E, C and E, D and E might be wrong.

(iii) To replace the symbol E, we first select the candidate set. This is composed of the legal patterns which agree with the symbols A, B, C and D with specified relations.

(iv) Among the legal patterns in the set, one symbol is selected by the criteria of the Hamming distance with the original symbol E, of the statistics, of the occurrence and, if possible, of the information about the quality of the original image. □

Fig.4-26 is an example of $(\text{LSC-picture})_2$ produced by the fixed mesh method. It can be seen that there are no illegal patterns, and the image quality appears not to be worse than that of the original image.



(a) original (b) (LSC-picture)$_2$

Fig.4-26 Example of a LSC-picture (fixed mesh method).

IV-2-5-3 Scanning Mesh Method

(LSC-picture)$_2$ is also produced directly from the original image. This method is to scan the window one by one on the pixel and at the same time to replace the illegal patterns by one of the legal patterns specified by the transcription function.

[Scanning unit mesh window method]
The window with the same size as the unit mesh moves one by one on the pixel like TV scanning, and at each position the following steps are executed.

(i) If the pattern in the window is a legal one, then the window is moved to the next position.

(ii) If it is an illegal pattern, it is temporarily replaced by the legal pattern specified by

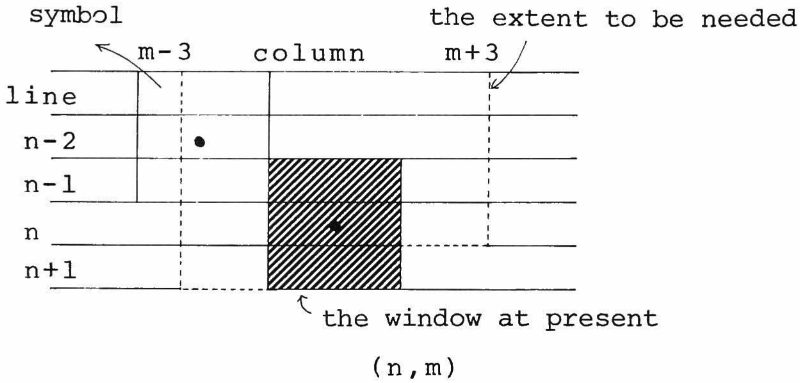


Fig.4-27 Scanning mesh method.

the transcription function.

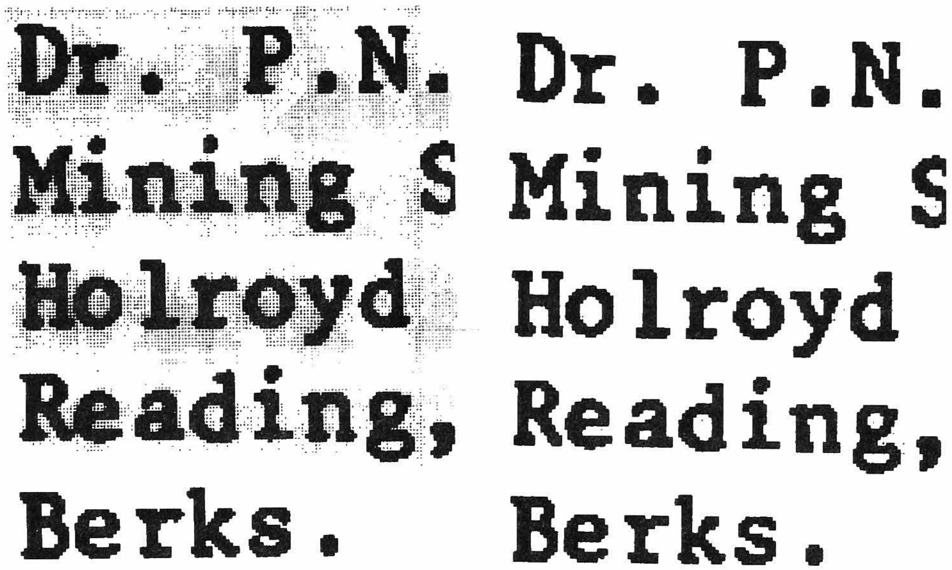
(iii) Then the backtrack process is called, i.e. as shown in Fig.4-27 the window backs to the extent of three lines and two columns, and checks whether or not an illegal pattern has been generated at any position due to the replacement.

(iv) If it has not generated any illegal pattern, the correction made in the step (ii) becomes permanent, and the window proceeds to the next position.

(v) If there occurs any illegal pattern due to the replacement, another legal pattern is selected and temporarily replaces the original illegal pattern. Then go to step (iii).

(vi) If the legal patterns are not found in the minimum Hamming distance, there is no problem in incrementing the distance. Then go to step (iii). □

Fig.4-28 is an example of $(\text{LSC-picture})_2$ produced by this method. Comparing this $(\text{LSC-picture})_2$ with that in Fig.4-26, there seems to be no difference as to the image qualities. Other items for comparison of these two methods are summarized in Table 4-4.



(a) original (b) $(\text{LSC-picture})_2$

Fig.4-28 Example of a LSC-picture (scanning mesh method).

IV-2-6 Consideration

IV-2-6-1 General Consideration on Methodology

MOLD Theory assures that LSC-pictures can be constructed for any line drawings satisfying the **Fundamental Assumption.** The processes of this transcription are considered from two standpoints.

The methodology taken here is the same as that of science in general. That is, first, we establish a hypothesis, i.e. construct the theory on some assumptions, and then various experiments are done to test the theory. These processes in our MOLD Theory are

Table 4-4 Comparison of two methods to make LSC-pictures.

	speed	memory capacity	quality	amount of hardware	advantage for the process
Fixed mesh method	B	6 lines	A	B	A
Scanning mesh method	A	5 lines	B	A	B

A : superior
B : inferior

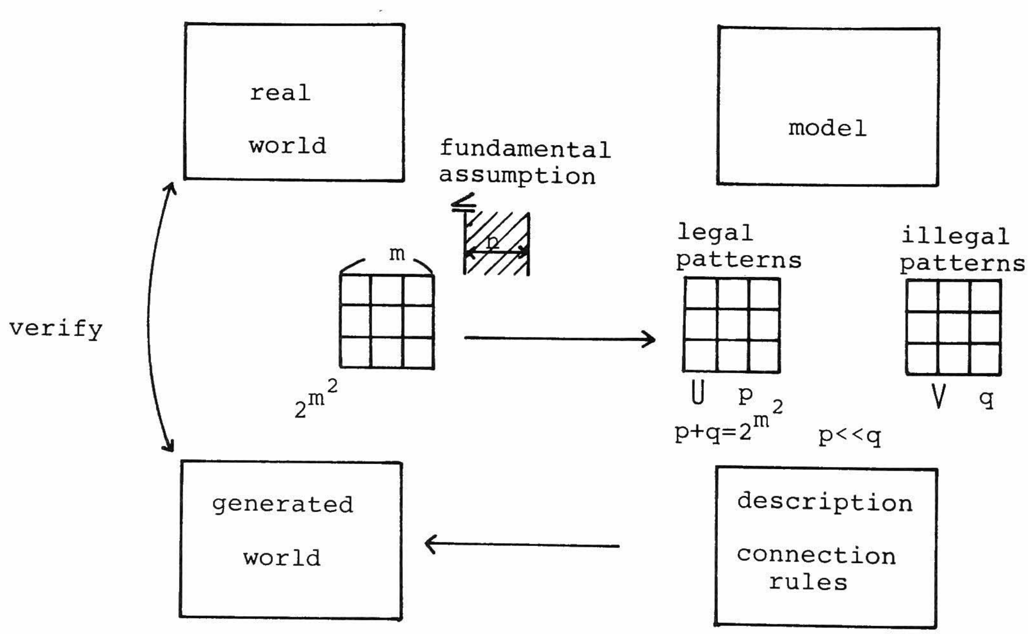


Fig.4-29 Processes of MOLD Theory.

illustrated in Fig.4-29. In this figure, the hypothesis is that if the **Fundamental Assumption** is satisfied in the line drawings' image, there cannot occur any illegal patterns except at the acute crossing points. This has been tested by observing the data. The methodology presented by us will be a powerful system for processing the various information of line drawings, although it has seldom been used in the field of information processing only because it is not wildely known.

Secondly, this theory seems to lead to an extention of the idea of filters. The transcription process is a kind of filter process which passes the legal patterns only while it cuts off the illegal patterns (see Fig.4-30). This filter handles the information of the

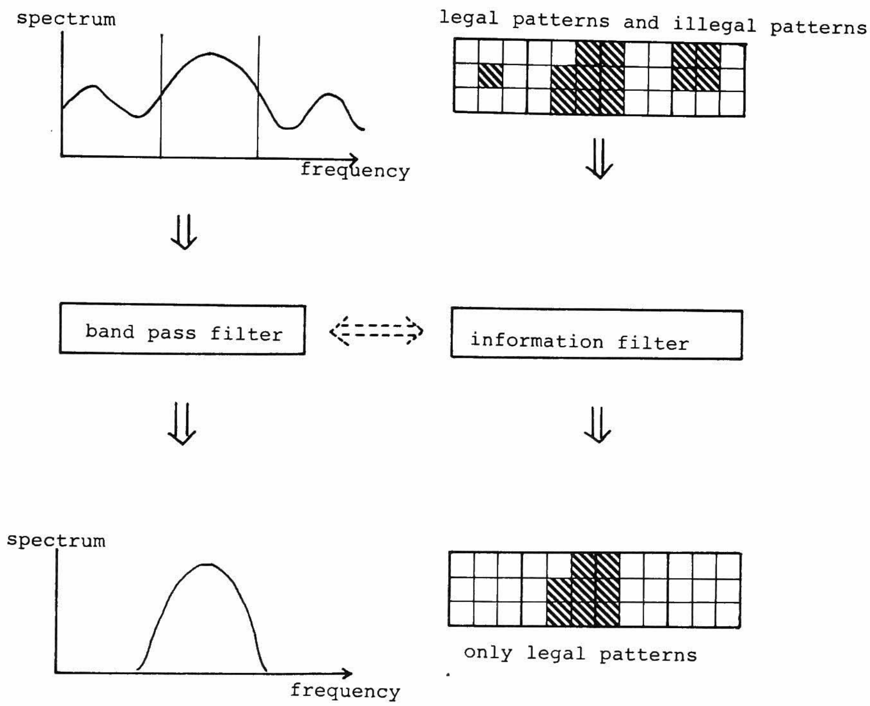


Fig.4-30 Concept towards an information filter.

pattern level data and the constraint of the lines i.e. of the tools to generate them. This concept is common to that described in CHAPTER III, and is also a very important proposal of this thesis.

IV-2-6-2 Fundamental Assumption

In MOLD Theory, the most important assumption is the **Fundamental Assumption** that the width of the digital line (n) is greater than or equal to the size of the unit mesh (m). We select this assumption because of its simplicity and the number of legal patterns. Indeed, if we select another assumption, we can construct another world.

For example, we consider the case that the assumption is $2n > m > n$ (where n is the width of the digital line and m is the size of the unit mesh). In this case the patterns in the window are restricted to only the following five cases.

(i) all white pixels
(ii) all black pixels
(iii) one white region and one black region
(iv) two white regions and one black region
(v) one white region and two black regions

We can define the legal patterns based on these five cases. In practice, however, m has both upper and lower limits, so that the selection of m is very difficult, especially when the image contains lines with various width.

Generally, if the size of the unit mesh is fixed, the number of legal patterns increases as the number of the digital lines which are included in the unit mesh. Consider the case that the width of the digital lines are relatively narrower; this is the same as the unit mesh size becoming greater and the constraint ratio becoming higher. To consider an actual application, it is preferable that the number of legal patterns is small.

In treating the graphs, it will be efficient to select the size of the unit mesh between the width of the digital line and the smallest symbols in the graphs. For special purposes, a modification of our theory may be possible.

IV-2-6-3 Width of Lines

In the condition of the **Fundamental Assumption,** the lower limit of the digital line is specified by the size of the unit mesh. It is convenient not to have an upper limit in practical use, but if the line width becomes much greater than the size of the unit mesh, it is easy to imagine a drop in the efficiency. In section IV-4, it is shown that the most efficient line width is almost equal to the size of the unit mesh.

Another factor concerning the width of the digital line is the sampling interval. The development of techniques in the field of semi-conductors will make our sampling condition become more practical.

IV-2-6-4 Effects of Shift in Mesh Division

The position of the mesh division in the fixed mesh method may cause different corrections to illegal patterns. In the fixed mesh method, illegal patterns are eliminated either by the process of making a SYM-picture or by the process of making an LSC-picture from the SYM-picture. The same illegal patterns therefore may be corrected differently, as shown in Fig.4-31.

In the scanning mesh method, this kind of problem does not occur. A difficult problem in this method is that the process has a tendency to correct a larger region. An example is shown in Fig.4-32. In this figure, the two upper black pixels must be turned to white, but the process deletes the black pixels of the lower region, which causes a lot of corrections.[1)]

Under these circumstances, there is a general problem with scanning, i.e. the result depends largely on the order of the processes. How to remove this problem is work for the future.

---

1) This correction is very much related to the order of scanning, and this example is one of the worst cases.

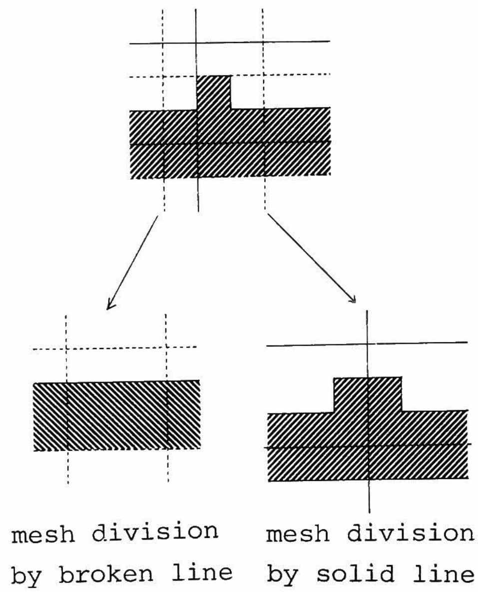


Fig.4-31 Correction of illegal patterns (fixed mesh method).

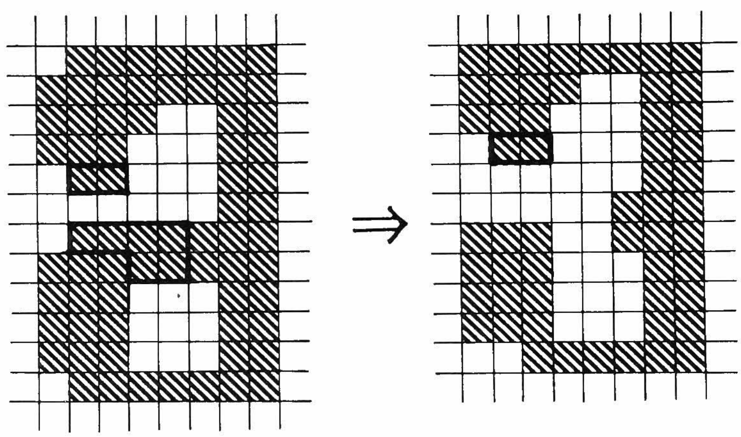


Fig.4-32 Correction of the illegal patterns
(scanning mesh method).

IV-2-6-5 Amount of Information

"What amount of information does a given image have?" We cannot generally answer this question. We can only say that the amount of information can be calculated if we assume a model of the information source. The amount of information greatly depends on the model. Therefore, the entropy of the image of one model is greater than the compression ratio of another model (see section IV-4).

Table 4-5 gives the values of the entropies in various models of the original images

Table 4-5 Comparison of LSC-pictures and the original images as to the amount of information contained.

	pixel memory less		pixel simple markov		pixel 2-D prediction		run-length memory less	
	original	LSC-picture	original	LSC-picture	original	LSC-picture	original	LSC-picture
CCITT NO.1	0.1708	0.1711	0.0669	0.0656	0.0355	0.0322	0.0595	0.0578
CCITT NO.2	0.2359	0.2357	0.0457	0.0453	0.0234	0.0227	0.0492	0.0486
CCITT NO.3	0.2806	0.2790	0.1142	0.1116	0.0522	0.0471	0.1073	0.1030
CCITT NO.4	0.4680	0.4667	0.2154	0.2110	0.1296	0.1176	0.2045	0.1981
CCITT NO.5	0.2823	0.2834	0.1216	0.1188	0.0606	0.0536	0.1155	0.1108
CCITT NO.6	0.2238	0.2240	0.0791	0.0762	0.0344	0.0309	0.0778	0.0740
CCITT NO.8	0.9803	0.9801	0.0927	0.0921	0.0379	0.0362	0.0916	0.0906

and their corresponding $(\text{LSC-picture})_2$, and Fig.4-33 shows the distribution of run-length between the images for CCITT NO.1 data. From this table the entropies of the $(\text{LSC-picture})_2$ are a little smaller than those of the original images in any other model except the pixel memory-less model. In the real world, when we store these images in memories, the necessary memory capacities are 9 to 7 in favor of LSC-pictures. We may conclude that the constraints of digital lines, i.e. information, are transformed to the amount of information, i.e. physical amounts.

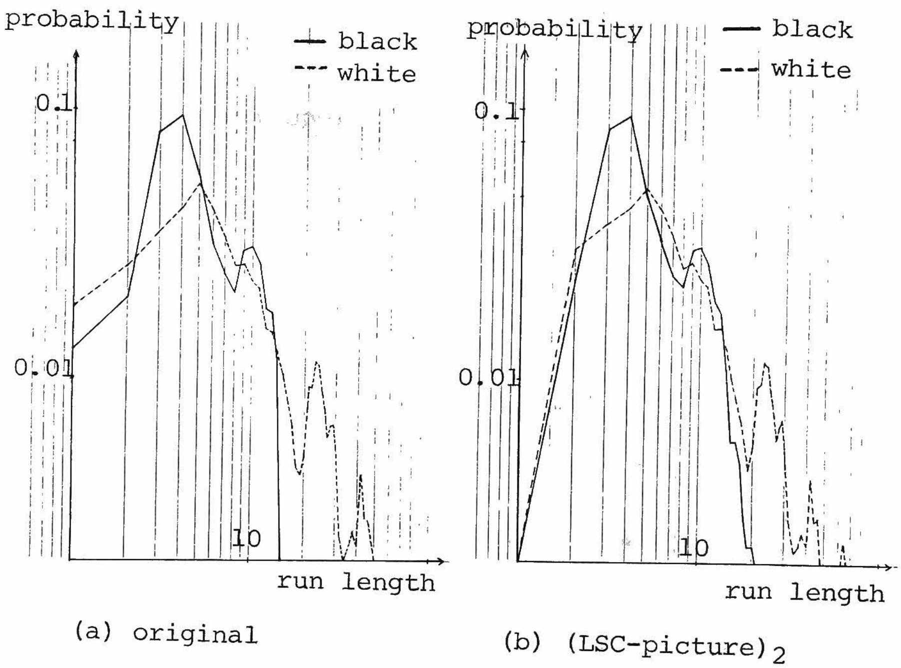


Fig.4-33 Distribution of run-length for a LSC-picture.

IV-3 Adaptive System for Noise Elimination
----- Application of MOLD Theory (1) -----

IV-3-1 The Problems

Nowadays the facsimile is considered to be useful equipment not only for electrical communication but also for the inputs of bi-level images to computers. Also facsimiles are essential devices for office automation (OA) or laboratory automation (LA). Considering this tendency, we must study the restrictions of the processed image in this device. In other words, bad quality images as well as good quality ones must be handled by computers.

The problem we discuss in this section is noise elimination for various image qualities. We cannot expect to have a good result from applying the same process to all of these various image qualities. The first task is to classify the input images according to their qualities. Three categories of image qualities are defined in order to select an adequate processing to the input images.

(1) "Blurred image", which contains broken lines due to noise, arising from very low contrast original documents, or high threshold setting for binarization of the facsimiles.

(2) "Stained image", which contains very much noise in various places on the paper. An image in this category is the result of too low a threshold setting for binarization, of very bad quality of the original documents, or of spots and stains contained in the original paper.

(3) "Good quality image", which is neither blurred nor stained. The processes reported by various reports have almost all restricted the input image data to this category.

After classification, the noise elimination processes are applied according to the category. The flow diagram of our adaptive system for noise elimination is shown in Fig.4-34.

Noise elimination for image processing has been well developed. To distinguish noise

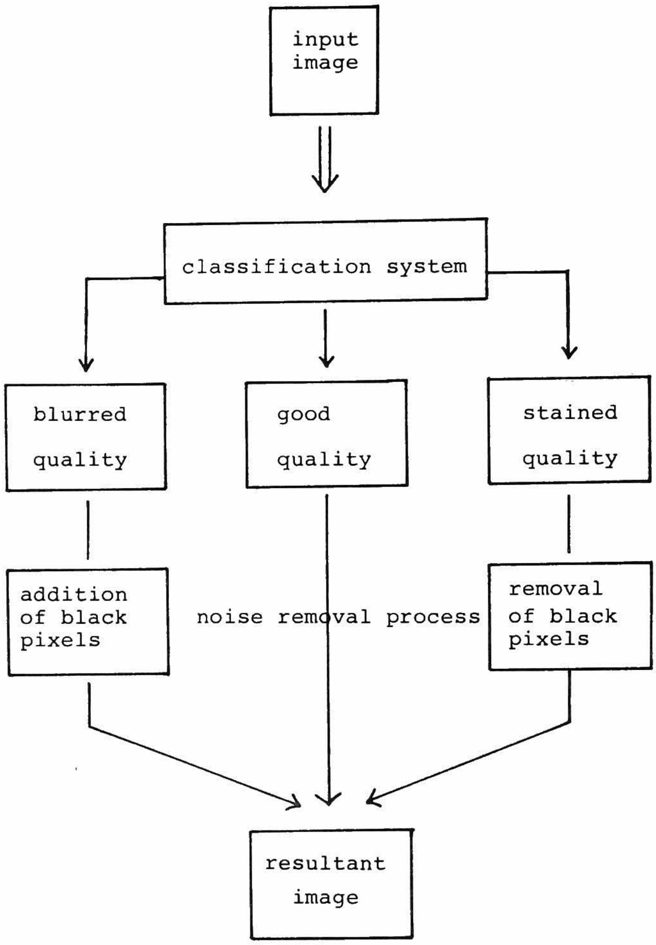


Fig.4-34 Flow of adaptive system for noise elimination.

from the image, it is necessary to have knowledge or information about that image, but this is often a difficult problem, because generally the noise generating process has a random nature. We devide the approaches into two classes: statistical approaches and heuristic approaches. The typical approach in the former class is to define models both for the images and the noise with their means and auto-correlation. Winer filter [24] and Kalman filter [25,26] are the most well known examples. It is true that the results of these approaches are good but the noise remaining is often prominent because they are based on statistics.

To the contrary, in the case that a priori knowledge is available to distinguish the image signal from the noise, an heuristic approach seems to be better. The approach developed by Rosenfeld [23] is very close to the concept of legal patterns. It is based on the assumption that the pictures in a certain local area have approximately constant gray level.

To restrict our discussion to bi-level images, the statistical method may not be applied because there are only two levels. The representative method in this field is called "propagation and shrinking" [16]. This technique is very general and the process is only applicable to the signal, so that if we restrict the world, we can do very well.

The main heuristics to distinguish the image signal from noise is considered to be the area of the black pixels [27], which costs large computation time. It is generally difficult to decide the threshold value to distinguish them.

The approach described in this section is basically the heuristic one above and the threshold values are decided by estimating the line width. If the image is blurred the lines are forced to be wider, while the black areas under the threshold are eliminated in the case the image is stained. Our system is adaptive to the key points that the process is selected according to the image qualities and the threshold to distinguish the signal from the noise is decided according to the line width included. The processes are basically used on the SYM-picture or the LSC-picture. In comparison with those on the original bi-level images, they will be expected to be faster and simpler.

IV-3-2 Classification of Input Images According to Their Qualities

IV-3-2-1 Three Classes of Image Qualities

In the case that the system must handle many kinds of input image with various properties, it is important to classify the inputs according to a certain property favorable to the process. Here, we classify the input images in order to decide how to process the input images, i.e. whether the image is positively transformed to decrease the black pixels, or to increase them, or the image quality is so good that the image must be handled symmetrically as to black and white. These three classes are called "blurred quality", "stained quality" and "good quality", respectively.

These three classes are not divided purely quantitively. The boundaries among them are not explicit and vary slightly according to who observes them. It is not important which class the images located around these categories' boundaries are in, as long as the purpose of this classification is only to obtain rough information of the qualities.

The criteria of each class are as follows:

(1) "blurred quality" --- the blur of the lines or the lack of the lines.
(2) "stained quality" --- isolated points of noise located in the background.
(3) "good quality" --- the above mentioned features are not prominent.

Fig.4-35 is an example of various kinds of image qualities. The first row illustrates the class "blurred" and the second row does "good" and the third row does "stained". They are generated by means of copies from the blue copies. One example of the input images obtained from the original on cheap, thin paper is shown in Fig.4-36, and it seems to be almost the same as those in the third row of Fig.4-35.

In practice, the input images have noise not only uniform over the whole area but also concentrated in some parts. The problem of classifying the input images, therefore, must be considered to be a local area affair, kept as small as possible. However, the classification is easier as the objective images become larger. We cannot decide the optimum size of the area for this trade-off problem because the evaluation function for each factor is not known. Hence, the size of the area is decided emprically to be 255 × 255 pixels of the image, i.e. 85 × 85 symbols of the SYM-picture or the LSC-picture. It is about one sixtieth as large as the whole image of A4 size as sampled in the fine detail mode of the facsimile device.

IV-3-2-2 Properties of Legal Patterns

The legal patterns can be divided into some classes according to their properties. They are again shown in Fig.4-37 which is the same as Fig.4-13. First, they are divided into 14 classes of the configuration of black and white pixel patterns without considering rotation. The symmetry patterns, as to black and white pixels, are grouped together, giving 7 classes which are named A, B, C, D, E, F and G, respectively. They are shown in Fig.4-38.

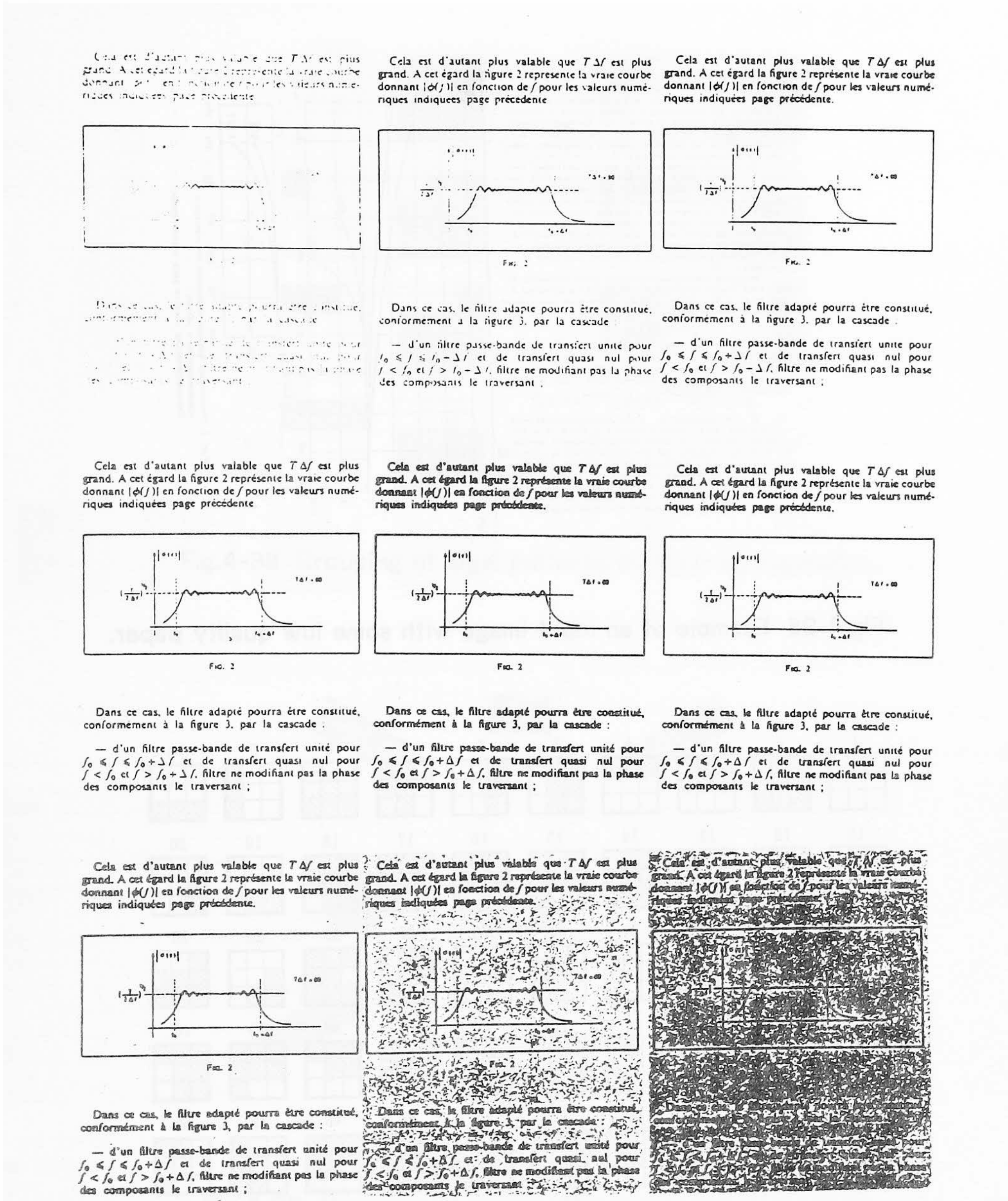


Fig.4-35 Examples of various image qualities.

Next, to the contrary, the legal patterns except one with all white pixels (W pattern) are devided into 7 classes by consideration of the relative locations between a digital line and the window i.e. the possible direction of each legal pattern (see Fig.4-39).

For the groups 1, 2, 3 and 4 in Fig.4-39, the relations shown in Fig.4-40 (this is an

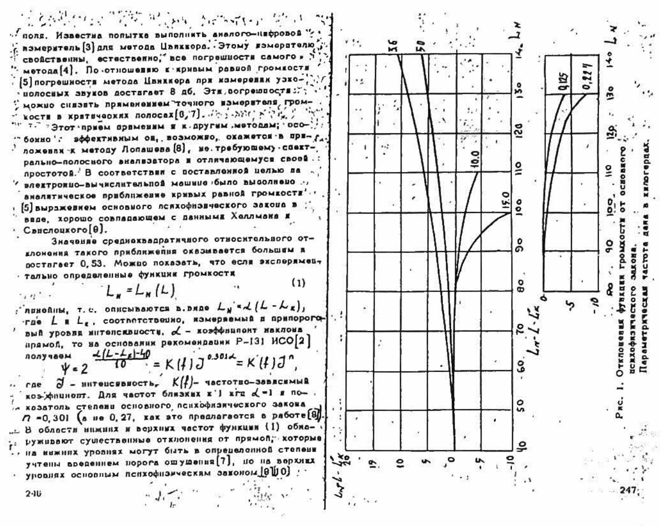


Fig.4-36 Example of an input image with some low quality paper.

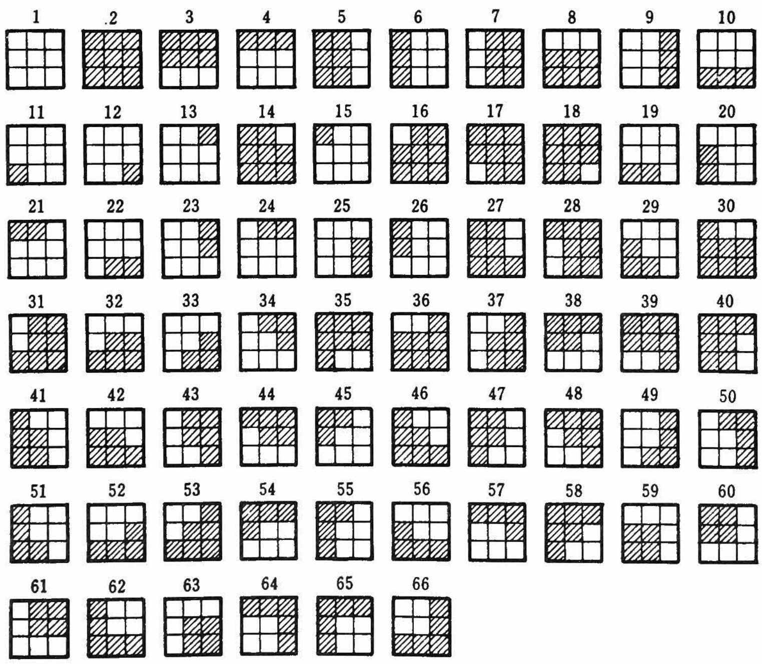


Fig.4-37 Legal patterns (m=3).

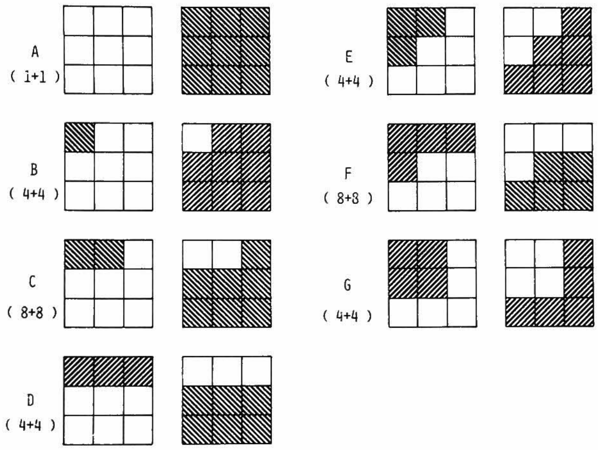


Fig.4-38 Grouping of legal patterns by their configuration.

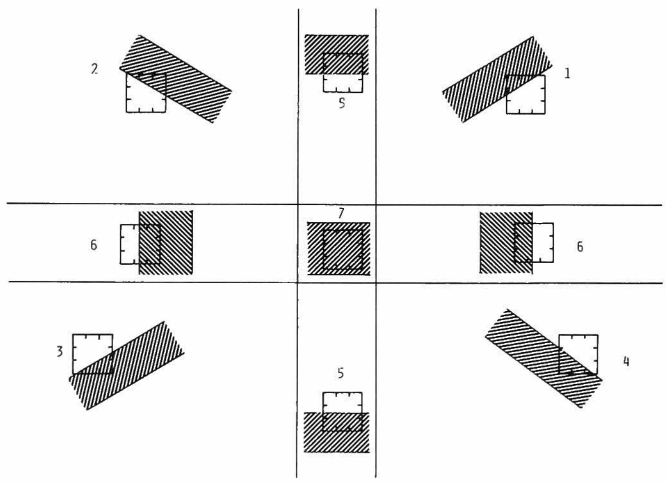


Fig.4-39 Grouping of legal patterns by their direction.

example in group 4) are exactly established in an LSC-picture. This fact is verified from Table 4-6 which gives the statistics of the LSC-picture in the case of CCITT's test chart NO.1. These relations are based on the assumption of overlapped windows i.e. in the case (1) in Fig.4-40, each of the legal patterns on the left side of the equation occurs in the position just to the left by one pixel of the legal pattern on the right side of the equation (see Fig.4-41).

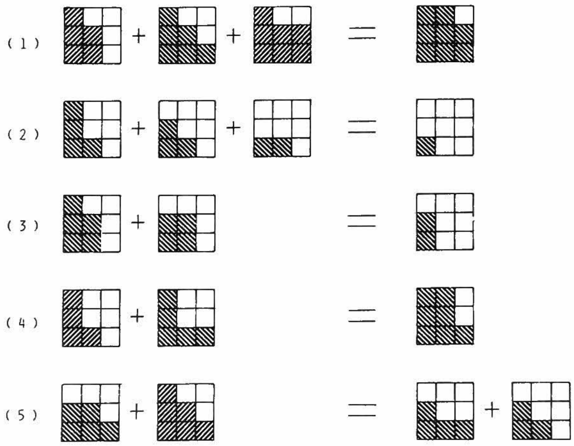


Fig.4-40 Relations among legal patterns.

Table 4-6 Statistics of the LSC-picture corresponding to CCITT test chart NO.1.

num	population	num	population	num	population	num	population	num	population	num	population
1	3,840,551	2	46,417	3	8,169	4	8,169	5	7,673	6	7,673
7	7,383	8	7,174	9	7,383	10	7,174	11	5,153	12	5,121
13	4,854	14	4,584	15	4,565	16	4,552	17	4,285	18	3,996
19	2,418	20	2,609	21	2,363	22	2,369	23	2,523	24	2,284
25	2,443	26	2,324	27	2,040	28	1,954	29	1,500	30	1,849
31	1,874	32	1,638	33	1,489	34	1,318	35	1,794	36	1,800
37	1,712	38	1,341	39	1,715	40	1,755	41	1,631	42	1,440
43	1,517	44	1,278	45	1,058	46	1,104	47	1,302	48	1,053
49	1,263	50	1,252	51	1,235	52	1,189	53	1,040	54	1,183
55	1,144	56	1,044	57	1,013	58	900	59	978	60	1,022
61	1,006	62	805	63	731	64	702	65	611	66	611

It is investigated how the occurrence statistics of legal patterns are deflected as the image qualities vary. The results are shown in Fig.4-42. The input images for the experiment are CCITT's test chart NO.1 (English letter) and sampled by the facsimile in the fine detail mode. The samples are shown in Fig.4-43. The image size is 1,728 × 2,352 and the scanning window is moved one pixel at a time, so that the total sample number attains 4,056,100.

Here, we consider the results of Fig.4-42 as the important information of the groups of legal patterns, i.e. Fig.4-38 and Fig.4-39 and of the relations between them i.e. Fig.4-40. It will be mentioned below that the order of the occurrence of the legal patterns is not ad hoc but universal.

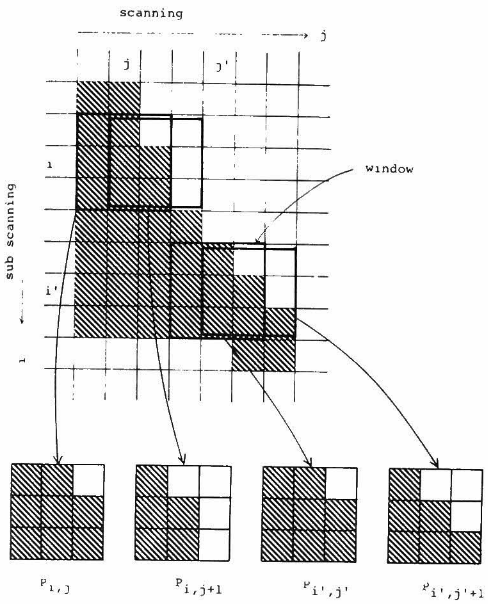


Fig.4-41 Explanation of the relation (1) in Fig.4-40.

Among the legal patterns in the group D of Fig.4-38, the occurrence frequency for the good quality images is flat while those of the other qualities are not flat. This means that the digital lines are somewhat disturbed. In particular, the number of  patterns is almost the same as that of  patterns for the good quality images. The same equations are established in each rotation. In the statistics of "blurred" images the number of  patterns is greater than that of  patterns. This is because the digital lines are shifted to be thin down to about one pixel's width. In the statistics of the "stained" image, the deflection depending on the direction is prominent, which may be due to the property of the noise.

The histogram of the group B in Fig.4-38 has more peaks and valleys in the statistics of "blurred" or "stained" images than those of "good quality". Noise generally increases the kinds of  and  patterns very much. In the case of "blurred" images,  patterns decrease greatly because the digital lines are very thin, while in "stained" images,  patterns increase greatly due to the noise in the background. The group C in Fig.4-38 shows the same inclination.

The groups shown in Fig.4-39 do not appear to be as powerful for the classification according to image qualities, though, they have the possibility of being a powerful parameter if we classify the input images by the kinds of the character patterns i.e. alphabets, chinese characters, etc.

The relations presented in Fig.4-40 play an important part in classification and are

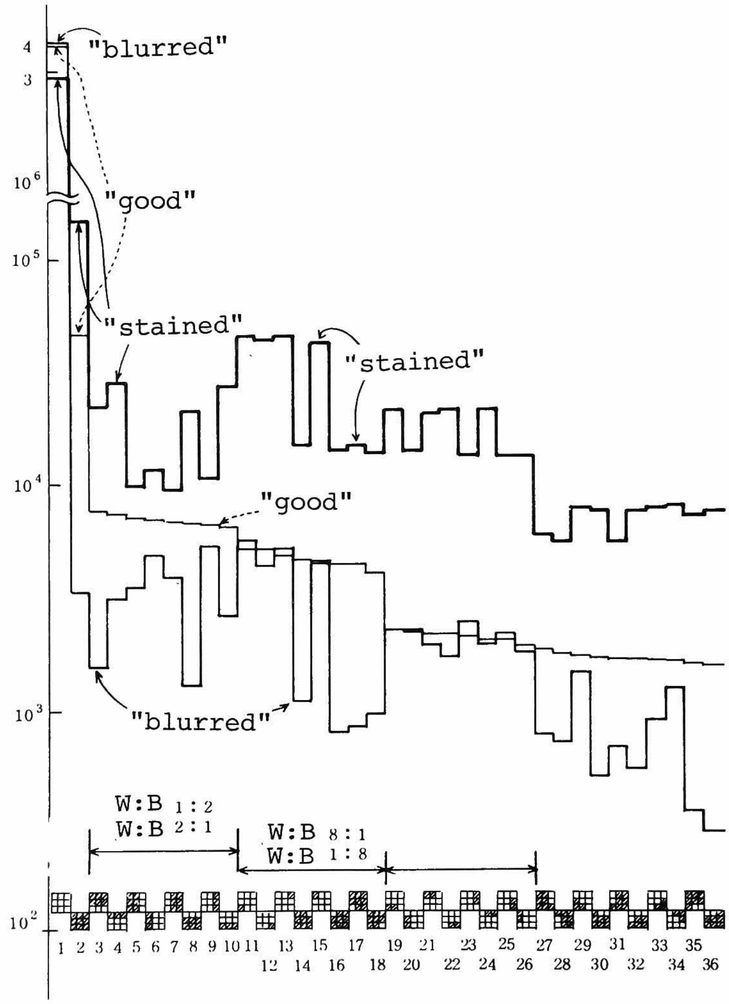
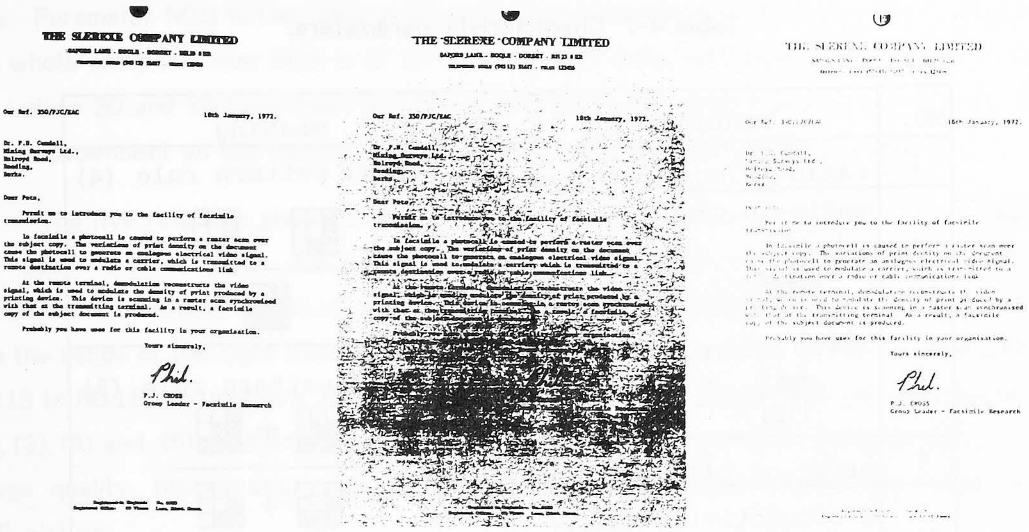


Fig.4-42 Statistics of the legal patterns in three kinds of image qualities.

almost satisfied in the case of "good quality" images while not in other cases.

IV-3-2-3 Definition of Characteristic Parameters for Image Quality

The considerations above bring us to define the characteristic parameters for classification. The condition to be satisfied in our system is to classify only one bi-level



(a) "good" (b) "stained" (c) "blurred"

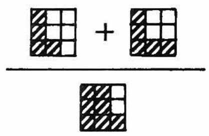
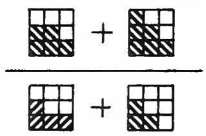
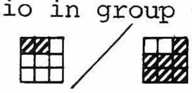
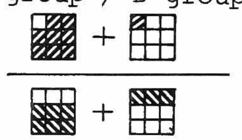
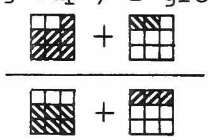
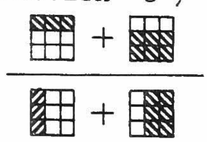
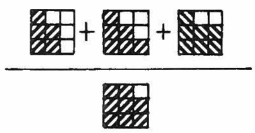
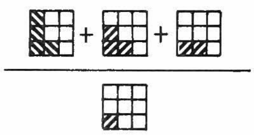
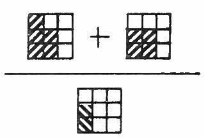
Fig.4-43 Input images for the statistics.

image. It is impossible to compare the input image with the original native good quality image. And the percentage of black pixels varies greatly because we have no a priori knowledge for the input images except that of the line drawings. We must, therefore, be very careful when introducing the parameters representing that percentage. Still more, this process is located at a pre-process, so that a simple algorithm is necessary.

The characteristic parameters defined here are based on the statistics of the legal patterns in 255 × 255 pixel images and summarized in Table 4-7. Parameter NO.1 is the ratio of the legal patterns to all possible patterns except W pattern. The more the digital lines contained are disturbed, the lower the value of this parameter. The W pattern is removed from the calculation of this parameter in order to make the parameter independent of the amount of black pixels (i.e. the digital lines) on the paper.

Parameters NO.2 and NO.3 are the ratios in the group B and C, respectively. These parameters become extremely large when the image becomes "blurred", because and patterns under the bar in Table 4-7 decrease in the "blurred" image. In the case of "good quality" image, the occurrence numbers of these patterns are about the same, while the patterns both under and above the bar increase by about the same percentage in the case of "stained" images. (This fact is easily deduced from Fig.4-42.)

Table 4-7 Characteristic parameters.

NO.	meaning	NO.	meaning
1	ratio of legal patterns	19 – 22	legal pattern rule (4) 
2	ratio in group B 	23 – 26	legal pattern rule (5) 
3	ratio in group C 	27	B group / D group 
4	order of black pattern	28	C group / D group 
5	W connectivity (line)	29	direction 5 / 6 
6	W connectivity (space)		
7 – 10	legal pattern rule (1) 		
11 – 14	legal pattern rule (2) 		
15 – 18	legal pattern rule (3) 		

Parameter NO.4 in Table 4-7 is the ordinal number of the all-black pattern in the statistics. In the case of "good" or "stained" images, this ordinal is secondary because the all-black pattern occurs in the digital lines while the other legal patterns except W pattern appear at the boundary part of the digital lines. (This is somewhat dependent on the width of the digital lines.)

Parameters NO.5 and NO.6 represent the ratio of the white large area to the whole one. Parameter NO.5 is the ratio of 30 continuous W patterns in the scanning direction to the whole and parameter NO.6 is of 15 × 15 compact areas consisting of W patterns. Here, the values 30 and 15 have been decided from several pre-experiments. These ratios are mainly dependent on the amount of digital lines included in the objective images.

The five relations shown in Fig.4-40 are available for the characteristic parameters. For each relation, there are four cases which correspond to 1, 2, 3 and 4 in Fig.4-39. Parameters from NO.7 to NO.10 correspond to the first relation (1) in Fig.4-40, and they are the ratios of the right expression to the left. Parameters NO.11 to NO.14, parameters NO.15 to NO.18, parameters NO.19 to NO.22 and parameter NO.23 to NO.26 correspond to (2), (3), (4) and (5) relations in Fig.4-40, respectively. According to the degration of the image quality, these ratios deviate from the value 1.0, which means the image is an LSC-picture.

Parameters NO.27 and NO.28 are the ratios of the number of the patterns in B and C groups to that in D group, respectively. The legal patterns in B and C groups represent the case that the digital lines are neither in vertical nor in horizontal directions. If there is some relation between the straightness of the digital lines and the degration of the image quality, these parameters will be useful.

Parameter NO.29 is useful for measuring the direction of the legal patterns in D group. The degration of the image quality may affect this parameter as in the case of the statistical data in Fig.4-42, particularly in the "stained" images.

IV-3-2-4 Experimental Evaluation of the Characteristic Parameters

To evaluate the usefulness or features of these characteristic parameters, we consider the F-ratio of the variance between the image-quality classes and those in each image-quality class. The F-ratio for the k-th parameter is defined as follows:

$$F_k = \frac{1/m\sum_{r=1}^{m}(X_r (k) - X(k))^2}{1/(n-1)m\sum_{r=1}^{m}\sum_{j=1}^{n}(X_{rj} (k) - X_r (k))}$$

where $X_{rj}(k)$: value of k-th parameter for j-th sample in the image-quality class r

$X_r(k)$: $1/n\sum_{j=1}^{n}X_{rj}(k)$ i.e. the average of k-th parameters in the image-quality class r

$X(k)$: the average of k-th parameters in all the classes

n : the number of samples

m : the number of classes

We select approximately the same quality of image in the middle one of each column in Fig.4-35. The data are obtained from CCITT's test charts NO.1, NO.2, NO.4 and NO.5 (shown in Fig.4-8). Forty images with size 255 × 255 pixels from each data set are selected. Therefore, we have about 160 images for each class of image quality.

From these images, the characteristic parameters are calculated in each class, and the statistical data are obtained. To investigate which parameter is powerful enough to

Table 4-8 F-ratio of the characteristic parameters.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
b-g-s	0.7464	0.2288	0.1949	1.0753	0.7230	0.6438	0.6651	0.3686	0.3103	0.8805	1.1367	1.1535	1.0862	1.2213	1.0035
g-s	1.0528	0.1506	0.2211	0.0060	0.5717	0.7000	0.1191	0.1906	0.0618	0.2148	2.3934	2.0640	2.0714	2.1334	1.5398
b-g	1.1321	0.3320	0.2219	1.1681	0.0236	0.0026	0.6544	0.2957	0.2648	0.7791	0.8154	0.7839	1.0002	0.9450	1.3434
b-s	0.0019	0.0679	0.0705	0.9026	0.9524	0.9617	0.4099	0.2579	0.2084	0.5705	0.1345	0.2038	0.0638	0.1919	0.0025

b - "blurred" g - "good" s - "stained"

	16	17	18	19	20	21	22	23	24	25	26	27	28	29
b-g-s	0.9428	1.0530	0.8777	0.7413	0.4437	0.4509	0.4149	0.6003	0.5853	0.6909	0.7360	0.2838	0.2783	0.0083
g-s	1.4996	1.5543	1.3927	0.2417	0.1970	0.2132	0.2678	0.7593	0.7906	0.6191	0.8780	0.3794	0.4196	0.0061
b-g	1.2441	1.4457	1.1576	0.6502	0.3717	0.3892	0.3458	0.8355	0.7711	0.9737	1.0814	0.1241	0.1681	0.0086
b-s	0.0015	0.0051	0.0003	0.4783	0.2969	0.2918	0.2775	0.0262	0.0503	0.1325	0.0642	0.1140	0.0792	0.0038

decide one of the image classes uniquely, the F-ratios between two classes are calculated for all the combinations of image classes. They are summarized in Table 4-8.

The results show that the available parameters which distinguish the "good quality" image from the others seem to be parameter NO.1, parameters NO.11 to NO.14 and parameters NO.15 to NO.18. The averages of parameters NO1, NO.11 and NO.15 for the image qualities shown in Fig.4-35, except the center image, are plotted in Fig.4-44 to demonstrate the above fact. From this figure, the configurations of these curves have peaks near "good quality". The parameters, except NO.1, are based on the relations (2) and (3) in Fig.4-40. The reason why these relations are powerful is considered to be as follows. The relation (2) includes patterns and the relation (3) does patterns. From the statistics displayed in Fig.4-42, these two

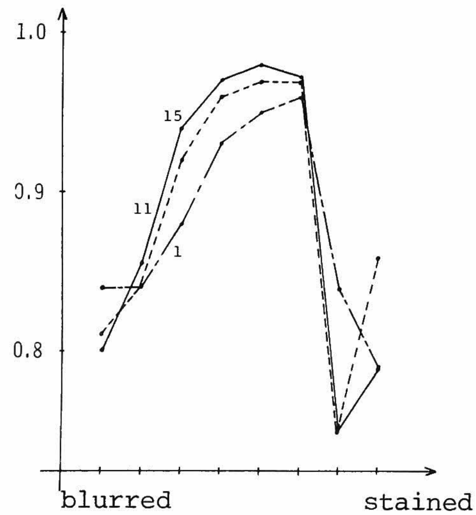


Fig.4-44 Transition of the parameters to distinguish "good quality" from others.

patterns have the property of increasing as the degree of degration of the image quality. Owing to this property, the parameters based on the relations (2) and (3) can distinguish the "good quality" images from the others.

On the other hand, parameters NO.4, NO.5, NO.6, NO.7 to NO.10 and NO.19 to NO.22 are important to distinguish between "blurred" images and "stained" images as seen from Table 4-8. These fact is verified by the graphs of the average values of the parameters NO.4, NO.5, NO.6, NO.7 and NO.19 They are shown in Fig.4-45 in the same fashion as in Fig.4-44. These curves are almost monotonic in the degree of gradual transition from "blurred" images to "stained" ones. It is easy to see why the parameters NO.4, NO.5 and NO.6 are available for this purpose. The parameters NO.7 to NO.10 and NO.19 to NO.22 are based on the relations (1) and (4) in Fig.4-40. The relation (1) includes patterns and the relation (4) does patterns. These two patterns are not so frequently observed in "blurred" images as the others, which is verified in Fig.4-42. This is the reason why these parameters are useful to separate the "blurred" images from the "stained" ones.

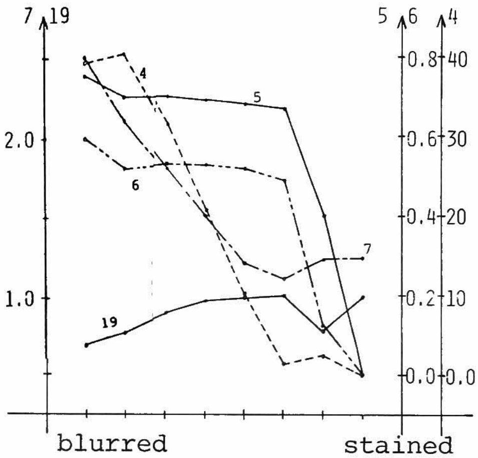


Fig.4-45 Transition of the parameters to distinguish "blurred" and "stained".

The other parameters are not found to be useful from this experiment, but the parameters NO.2 and NO.3 have features for the "blurred" images, in other words, they become relatively large for the "blurred" image. This is also deduced from the statistical data shown in Fig.4-42. Therefore, these parameters are only to be referred to in the case that the image may be "blurred".

Parameters NO.23 to NO.29 seem not to be useful for this purpose, but we cannot deny the possibility that in the case of other classifications, for example, according to various kinds of character patterns, some of these parameters will turn to be very useful.

IV-3-2-5 Classification System

The actual classification system is constructed on the PANAFACOM U-300 computer system with the main memory, 64 Kbyte. The cycle time of this computer is 650 ns, and the input image is stored in a disk. The program was written in FORTRAN and it takes about 80 seconds to classify the input image with size 255 x 255 pixels.

The configurations of the parameters which are taken to be efficient are interesting. From Fig.4-44, the "good quality" images may easily be separated from the others. The configuration of the parameters in Fig.4-45 shows the difficulty of the distinction between "blurred" and "stained" images unless the "good quality" images are removed.

Based on this fact, the strategy of the classification is decided. For good results we must introduce a hierarchical classification rather than to classify three classes at the same time. In other words, the system first decides whether the input images belong to the "good quality" or not. Only to the images decided not to be of "good quality", is the process to distinguish between "blurred" and "stained" applied (see Fig.4-46).

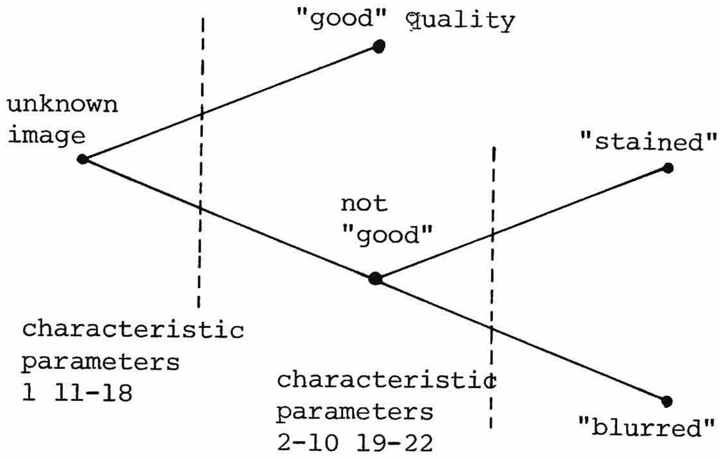


Fig.4-46 Strategy of image classification.

The system calculate the 22 parameters (i.e. NO.1 to NO.22) for the input images. First, the parameters NO.1, NO.11 to NO.18 are evaluated. If an input image is classified as "good quality", the process is finished. Otherwise, the second test is applied in order to decide whether "blurred" or "stained" by the parameters NO.2 to NO.10 and NO.19 to NO.22. The parameters NO.2 and NO.3 contribute only to the "blurred" images while NO.4 does only to the "stained" images.

The evaluation of each parameter is done independently. The calculated values of the parameters in the input images are compared with the average values of the parameters in one class. In other words, the difference of the two parameters is evaluated by the average of the Euclid distance and Mahalanobis distance. The score of the likelihood of belonging to that class is the weighted average of the efficient parameters. The weights are selected proportional to the F-ratios, and if the score for a class is greater than the threshold value already given emprically, the image is decided to belong to that class.

The average values of the parameters are calculated for the images having nearly the same qualities as shown in Fig.4-35. The samples of each class are about 120. The average and the variance of them are also calculated.

The errors of the classification must be considered. The errors occurring in the first classification are not so critical, while those occurring in the second decision are harmful to the following processes, because the directions of the process are contrary to each other. Therefore, in our system, the reject ratio (i.e. the threshold to decide the classes) of the second decision is settled rather high to avoid the classification errors.

IV-3-2-6 Experimental Results

Experimental data are selected from the images which are used in taking the average of the parameters. Strictly speaking, they are not the same because the positions are different and they are again digitized by the facsimile. Also 20 data are selected from different images. The total number of the experimental data is 165. The results are summarized in Table 4-9. The correct answers are decided by human inspection.

Table 4-9 Experimental results of classification.

		distinction of experimental system			
		blurred	good	stained	reject
correct answer (by man)	blurred	36	1	0	8
	good	0	53	0	4
	stained	4	0	43	16

The recognition ratio of the system was 98.5% and the reject ratio was 17.0%. The reject ratio became rather high in order to decrease the errors in the second decision as much as possible. This is natural since the purpose of the system is only to obtain information about the direction of the process. Therefore, the rejects occurred almost all in the second classification. This fact, in a sense, shows a little loss, i.e. the system knows that the input image was not "good quality", but cannot decide whether "blurred" or "stained". This is because the parameters to be used at the second classification are not really powerful.

This fact is verified from another fact. Four fatal errors occurred at the second

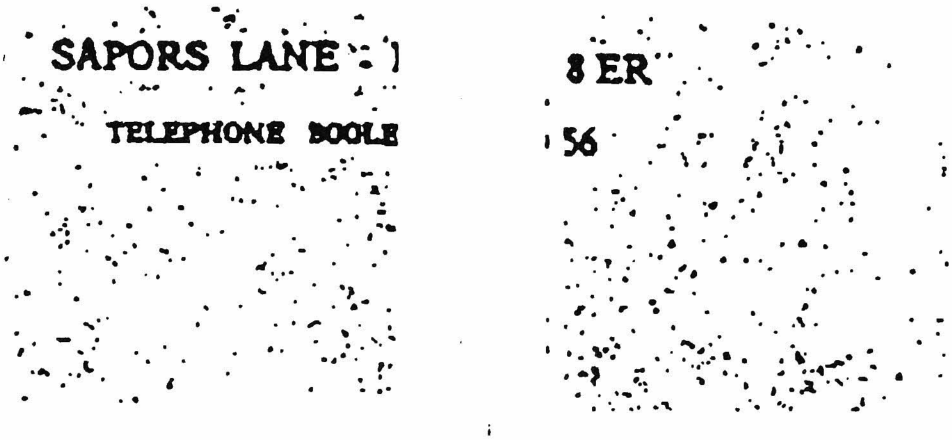


Fig.4-47 Examples of classification error images.

classification. Two of these are shown in Fig.4-47. They were classified as "blurred" class though they are "stained", but one can see that the figures do not include enough digital lines. The reason why these errors occurred is in the parameters NO.5 and NO.6 which had large weights at classification. They are largely dependent on the amount of digital lines included.

As a whole, the system can be said to be useful in practice. Based on this information, the adaptive system for noise elimination is constructed. The detail is described in the next section.

IV-3-3 Noise Elimination System

IV-3-3-1 Transcription Tables for SYM-pictures

As described in section IV-2, a transcription function is necessary to make SYM-pictures. In other words, if an illegal pattern is detected, which legal pattern to replace it must be decided. The result of the classification system according to the image quality is available for this decision.

Transcription table consists of 2^9 entries, and each corresponds to the possible pattern in the 3×3 unit mesh. In the column of each entry, the number of the legal pattern to be replaced is stored.

We prepare five transcription tables according to the classified image quality, one for "stained", one for "good quality", two for the rejected images and the other for "blurred". These translation tables are derived by the following process. The complete lists of them can be seen in Table 4-10.

(1) For "stained" images
The candidates' set of legal patterns is restricted to those members which have less black cells than the illegal pattern to be replaced. The legal patterns which are the nearest in Hamming distance are selected. If there are more than two legal patterns, we select the one which has the highest occurrence probability.

(2) For the rejected images likely to be "stained"
The candidates are the legal patterns with the minimum Hamming distance to the illegal pattern. If we have more than one candidate, the pattern with the least black cells is to be the replacing pattern.

(3) For "good quality" images
The legal pattern to replace the illegal one is simply selected on the basis of the Hamming distance and the occurrence probability.

(4) For the rejected images likely to be "blurred"
The strategy is the same as in case (2) except only the phase of "the pattern with the most black cells".

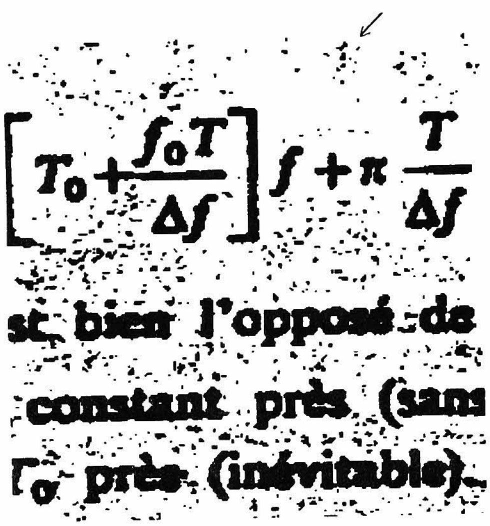
(5) For the "blurred" images
The candidates' set is restricted to the legal patterns with more black cells than the illegal one. The other starategy is the same as that in (1).

The rejected images have the feature such that neither of the scores for "stained" nor "blurred" is greatly superior to the other. To the transcription, even this ambiguous information seems to work reasonably well, i.e. the higher score selects whether (2) or (4). Though, the tables (2) to (4) are almost the same in practice. Local noise is reduced largely in this transcription. One example is shown in Fig.4-48. They are seen to be almost the same but 477 pixels turn to white due to the translation table (1). This is because the noise is only partly reduced as marked in Fig.4-48.

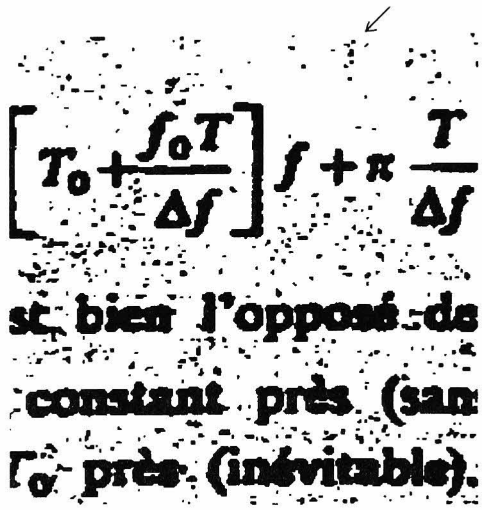
Table 4-10 Transcription table.

legal pattern

	(5)	(4)	(3)	(2)	(1)																										
0000	1	1	1	1	1	0001	12	12	12	12	12	0002	19	19	1	1	1	0003	22	22	22	22	22	0004	11	11	11	11	11
0005	10	10	10	11	11	0006	19	19	19	19	19	0007	10	10	10	10	10	0008	23	23	1	1	1	0009	25	25	25	25	25
000A	33	33	33	1	1	000B	33	33	33	33	33	000C	52	19	11	11	11	000D	52	52	25	25	25	000E	52	52	19	19	19
000F	52	52	52	52	52	0010	19	11	1	1	1	0011	63	22	12	12	12	0012	59	19	1	1	1	0013	63	63	22	22	22
0014	59	19	11	11	11	0015	32	10	10	10	11	0016	59	59	19	19	19	0017	32	32	10	10	10	0018	61	23	1	1	1
0019	63	63	25	25	25	001A	63	63	63	33	1	001B	63	63	63	63	63	001C	32	32	11	11	11	001D	32	32	32	25	25
001E	32	32	32	19	19	001F	32	32	32	32	32	0020	20	20	1	1	1	0021	56	20	12	12	12	0022	29	29	29	1	1
0023	56	56	22	22	22	0024	20	20	20	20	20	0025	56	56	20	20	20	0026	29	29	29	29	29	0027	56	56	56	56	56
0028	6	20	1	1	1	0029	8	9	25	25	25	002A	8	29	29	29	1	002B	8	8	33	33	33	002C	8	6	20	20	20
002D	8	8	8	20	20	002E	8	8	29	29	29	002F	8	8	8	52	52	0030	59	20	1	1	1	0031	42	42	12	12	12
0032	59	59	59	29	1	0033	42	42	42	22	22	0034	59	59	20	20	20	0035	42	42	42	20	20	0036	59	59	59	59	59
0037	42	42	42	42	42	0038	3	3	1	1	1	0039	8	8	8	25	25	003A	8	8	8	59	29	003B	8	8	8	63	63
003C	8	8	8	20	20	003D	8	8	8	32	20	003E	8	8	8	59	59	003F	8	8	8	8	8	0040	13	13	13	13	13
0041	9	9	9	12	12	0042	49	19	13	13	13	0043	49	49	22	22	22	0044	4	19	11	11	11	0045	66	9	9	9	11
0046	66	10	19	19	19	0047	66	66	10	10	10	0048	23	23	23	23	23	0049	9	9	9	9	9	004A	49	49	23	23	23
004B	49	49	49	49	49	004C	66	9	23	23	23	004D	66	66	9	9	9	004E	66	66	66	19	19	004F	66	66	66	66	66
0050	61	23	13	13	13	0051	37	9	9	9	12	0052	37	37	13	13	13	0053	37	37	37	22	22	0054	53	53	11	11	11
0055	53	53	53	9	9	0056	53	53	19	19	19	0057	53	53	53	10	10	0058	61	61	23	23	23	0059	37	37	9	9	9
005A	37	37	37	23	23	005B	37	37	37	37	37	005C	53	53	23	23	23	005D	53	53	53	9	9	005E	53	53	53	32	19
005F	53	53	53	53	53	0060	54	20	13	13	13	0061	49	9	9	9	12	0062	49	29	13	13	13	0063	36	49	22	22	22
0064	65	6	20	20	20	0065	36	56	20	20	20	0066	36	51	29	29	29	0067	36	36	56	56	56	0068	3	9	23	23	23
0069	36	49	9	9	9	006A	36	49	23	23	23	006B	36	36	49	49	49	006C	35	35	20	20	20	006D	36	36	9	9	9
006E	36	36	29	29	29	006F	36	36	36	66	66	0070	38	38	13	13	13	0071	36	36	9	9	9	0072	36	59	59	59	13
0073	36	36	36	37	22	0074	58	58	20	20	20	0075	36	36	36	42	20	0076	36	36	59	59	59	0077	36	36	36	42	42
0078	3	3	3	23	23	0079	36	36	9	9	9	007A	36	36	36	37	23	007B	36	36	36	37	37	007C	35	35	35	20	20
007D	36	36	36	8	9	007E	36	36	36	8	59	007F	36	36	36	36	36	0080	21	21	1	1	1	0081	50	21	12	12	12
0082	4	19	1	1	1	0083	7	10	22	22	22	0084	55	19	11	11	11	0085	50	10	10	10	11	0086	5	10	19	19	19
0087	27	52	10	10	10	0088	34	34	34	1	1	0089	50	50	25	25	25	008A	7	33	33	33	1	008B	7	7	33	33	33
008C	50	34	11	11	11	008D	31	50	25	25	25	008E	31	52	19	19	19	008F	31	31	52	52	52	0090	60	21	1	1	1
0091	43	43	12	12	12	0092	5	5	1	1	1	0093	7	7	7	22	22	0094	47	47	11	11	11	0095	27	27	10	10	10
0096	5	5	5	19	19	0097	27	27	10	10	10	0098	61	61	61	34	1	0099	43	43	43	25	25	009A	7	7	7	61	33
009B	7	7	7	63	63	009C	31	61	61	61	11	009D	31	31	31	32	25	009E	31	31	31	32	19	009F	31	31	31	32	32
00A0	45	45	45	1	1	00A1	50	45	12	12	12	00A2	5	29	29	29	1	00A3	27	56	22	22	22	00A4	55	55	20	20	20
00A5	27	55	20	20	20	00A6	5	5	29	29	29	00A7	27	27	56	56	56	00A8	3	34	34	34	1	00A9	39	50	25	25	25
00AA	3	3	29	29	29	00AB	7	7	33	33	33	00AC	35	55	20	20	20	00AD	8	8	8	20	20	00AE	5	5	29	29	29
00AF	14	8	8	8	52	00B0	60	60	60	45	1	00B1	27	60	60	60	12	00B2	5	5	5	59	29	00B3	27	27	27	42	22
00B4	47	47	47	20	20	00B5	27	27	27	42	20	00B6	5	5	5	59	59	00B7	27	27	27	42	42	00B8	3	3	3	60	34
00B9	39	39	39	43	25	00BA	3	3	3	59	59	00BB	14	7	7	7	63	00BC	35	35	35	47	20	00BD	14	8	8	8	32
00BE	14	5	5	5	59	00BF	14	14	8	8	8	00C0	24	24	24	24	24	00C1	50	50	24	24	24	00C2	7	4	24	24	24
00C3	7	7	7	22	22	00C4	65	4	24	24	24	00C5	31	50	24	24	24	00C6	31	31	19	19	19	00C7	31	31	10	10	10
00C8	34	34	34	34	34	00C9	50	50	50	50	50	00CA	7	7	34	34	34	00CB	7	7	7	49	49	00CC	31	50	34	34	34
00CD	31	31	50	50	50	00CE	31	31	31	34	34	00CF	31	31	31	66	66	00D0	61	61	24	24	24	00D1	43	43	43	24	24
00D2	7	7	7	24	24	00D3	7	7	7	37	22	00D4	58	58	24	24	24	00D5	31	31	31	43	24	00D6	31	31	31	19	19
00D7	31	31	31	7	10	00D8	61	61	61	61	61	00D9	43	43	43	43	43	00DA	7	7	7	61	61	00DB	7	7	7	7	7
00DC	31	31	61	61	61	00DD	31	31	31	43	43	00DE	31	31	31	7	61	00DF	31	31	31	31	31	00E0	54	54	24	24	24
00E1	38	50	24	24	24	00E2	38	54	24	24	24	00E3	7	7	7	22	22	00E4	65	65	65	20	20	00E5	16	65	65	65	20
00E6	40	40	29	29	29	00E7	16	16	16	56	56	00E8	3	3	34	34	34	00E9	39	39	50	50	50	00EA	3	3	34	34	34
00EB	16	7	7	7	49	00EC	35	35	34	34	34	00ED	16	16	16	50	50	00EE	16	16	16	29	29	00EF	16	16	16	31	66
00F0	38	38	38	24	24	00F1	39	38	38	38	24	00F2	40	38	38	38	38	00F3	16	7	7	7	37	00F4	58	58	58	38	20
00F5	16	16	16	58	38	00F6	40	40	40	5	59	00F7	16	16	16	27	42	00F8	3	3	3	61	61	00F9	39	39	39	43	43
00FA	16	3	3	3	61	00FB	16	16	7	7	7	00FC	35	35	35	3	61	00FD	16	16	16	31	43	00FE	16	16	16	31	3
00FF	16	16	16	16	16	0100	15	15	15	15	15	0101	4	21	12	12	12	0102	51	19	15	15	15	0103	62	10	22	22	22
0104	6	6	6	11	11	0105	62	6	6	6	11	0106	51	51	19	19	19	0107	62	62	10	10	10	0108	57	21	15	15	15
0109	64	9	25	25	25	010A	49	33	15	15	15	010B	28	49	33	33	33	010C	51	6	6	6	11	010D	30	52	25	25	25
010E	30	51	19	19	19	010F	30	30	52	52	52	0110	60	21	15	15	15	0111	46	46	12	12	12	0112	41	41	15	15	15
0113	46	46	22	22	22	0114	41	6	6	6	11	0115	46	46	6	6	6	0116	41	41	41	19	19	0117	46	46	46	10	10
0118	44	44	15	15	15	0119	48	48	25	25	25	011A	28	63	63	63	15	011B	28	28	63	63	63	011C	30	30	6	6	6
011D	30	30	30	32	25	011E	30	30	30	32	19	011F	30	30	30	32	32	0120	26	26	26	26	26	0121	62	6	26	26	26
0122	51	51	26	26	26	0123	62	62	62	22	22	0124	6	6	6	6	6	0125	62	62	6	6	6	0126	51	51	51	51	51
0127	62	62	62	62	62	0128	3	6	26	26	26	0129	30	30	25	25	25	012A	30	51	26	26	26	012B	30	30	30	33	33
012C	30	51	6	6	6	012D	30	30	6	6	6	012E	30	30	30	51	51	012F	30	30	30	62	62	0130	60	60	26	26	26
0131	46	46	26	26	26	0132	41	41	41	26	26	0133	46	46	46	41	22	0134	41	41	6	6	6	0135	46	46	46	6	6
0136	41	41	41	41	41	0137	46	46	46	46	46	0138	3	3	3	26	26	0139	30	30	30	25	25	013A	30	30	30	41	26
013B	30	30	30	8	63	013C	30	30	6	6	6	013D	30	30	30	8	6	013E	30	30	30	41	41	013F	30	30	30	30	30
0140	4	4	4	13	13	0141	64	4	4	4	12	0142	49	4	4	4	13	0143	28	49	22	22	22	0144	65	4	4	4	11
0145	62	62	4	4	4	0146	40	51	19	19	19	0147	46	62	10	10	10	0148	57	57	23	23	23	0149	64	64	9	9	9
014A	28	49	23	23	23	014B	28	28	49	49	49	014C	35	57	23	23	23	014D	48	64	9	9	9	014E	18	66	66	66	19
014F	2	53	66	66	66	0150	38	4	4	4	13	0151	48	48	48	4	4	0152	28	28	4	4	4	0153	28	28	28	37	22
0154	58	58	58	4	4	0155	46	46	46	4	4	0156	40	40	40	41	19	0157	2	46	46	46	10	0158	44	44	44	23	23
0159	48	48	48	9	9	015A	28	28	28	37	23	015B	28	28	28	37	37	015C	35	35	35	44	23	015D	2	48	48	48	9
015E	18	18	18	53	32	015F	2	2	53	53	53	0160	54	54	26	26	26	0161	38	54	26	26	26	0162	38	51	26	26	26
0163	17	62	62	62	22	0164	65	65	6	6	6	0165	46	62	6	6	6	0166	40	40	51	51	51	0167	2	46	62	62	62
0168	3	3	3	23	23	0169	39	39	9	9	9	016A	3	3	3	23	23	016B	17	17	17	49	49	016C	35	35	6	6	6
016D	2	2	2	6	6	016E	18	18	18	51	51	016F	2	2	2	30	62	0170	38	38	38	26	26	0171	39	38	38	38	26
0172	40	38	38	38	26	0173	17	17	17	46	37	0174	58	58	58	6	6	0175	2	46	46	46	6	0176	40	40	40	41	41
0177	2	2	46	46	46	0178	3	3	3	38	23	0179	39	39	39	3	9	017A	17	3	3	3	37	017B	17	17	17	28	37
017C	35	35	35	3	6	017D	2	2	2	30	3	017E	18	18	18	30	41	017F	2	2	2	30	30	0180	21	21	21	21	21
0181	64	4	21	21	21	0182	5	4	21	21	21	0183	27	27	21	21	21	0184	55	55	21	21	21	0185	27	55	21	21	21
0186	5	5	5	19	19	0187	27	27	10	10	10	0188	57	57	21	21	21	0189	64	64	64	21	21	018A	28	57	21	21	21
018B	28	28	28	33	33	018C	35	55	21	21	21	018D	14	64	64	64	21	018E	5	5	5	19	19	018F	14	14	14	52	52
0190	60	60	21	21	21	0191	48	48	21	21	21	0192	5	5	5	21	21	0193	27	27	27	21	21	0194	47	47	47	21	21
0195	27	27	27	47	21	0196	5	5	5	41	19	0197	27	27	27	5	10	0198	44	44	44	21	21	0199	48	48	48	43	21
019A	28	28	28	44	21	019B	28	28	28	7	63	019C	35	35	35	44	21	019D	14	14	14	48	32	019E	14	5	5	5	32
019F	14	14	14	27	32	01A0	45	45	45	45	45	01A1	27	54	45	45	45	01A2	5	5	45	45	45	01A3	27	27	27	45	45
01A4	55	55	55	55	55	01A5	27	27	55	55	55	01A6	5	5	5	51	51	01A7	27	27	27	62	62	01A8	3	3	45	45	45
01A9	39	39	39	45	45	01AA	3	3	45	45	45	01AB	14	14	14	33	33	01AC	35	35	55	55	55	01AD	14	14	14	55	55
01AE	14	5	5	5	51	01AF	14	14	14	27	62	01B0	60	60	60	60	60	01B1	27	27	60	60	60	01B2	5	5	5	60	60
01B3	27	27	27	5	60	01B4	47	47	47	47	47	01B5	27	27	27	47	47	01B6	5	5	5	5	5	01B7	27	27	27	27	27
01B8	3	3	3	60	60	01B9	39	39	39	3	60	01BA	14	3	3	3	60	01BB	14	14	14	27	3	01BC	35	35	35	47	47
01BD	14	14	14	27	47	01BE	14	14	5	5	5	01BF	14	14	14	14	14	01C0	4	4	4	4	4	01C1	64	64	4	4	4
01C2	28	54	4	4	4	01C3	28	28	4	4	4	01C4	65	65	4	4	4	01C5	48	64	4	4	4	01C6	40	40	4	4	4
01C7	2	2	2	4	4	01C8	57	57	57	57	57	01C9	64	64	64	64	64	01CA	28	28	57	57	57	01CB	28	28	28	64	64
01CC	35	35	57	57	57	01CD	2	48	64	64	64	01CE	18	18	18	57	57	01CF	2	2	2	28	64	01D0	38	38	4	4	4
01D1	48	48	48	4	4	01D2	28	28	4	4	4	01D3	28	28	28	7	4	01D4	58	58	58	4	4	01D5	2	48	48	48	4
01D6	40	40	40	5	4	01D7	2	2	2	27	5	01D8	44	44	44	44	44	01D9	48	48	48	48	48	01DA	28	28	28	44	44
01DB	28	28	28	28	28	01DC	35	35	35	44	44	01DD	2	2	48	48	48	01DE	18	18	18	28	44	01DF	2	2	2	28	28
01E0	54	54	54	54	54	01E1	39	38	54	54	54	01E2	40	38	54	54	54	01E3	17	17	17	54	54	01E4	65	65	65	65	65
01E5	2	58	65	65	65	01E6	40	40	40	65	65	01E7	2	2	2	27	62	01E8	3	3	3	54	54	01E9	39	39	39	64	64
01EA	17	3	3	3	54	01EB	17	17	17	28	64	01EC	35	35	35	65	65	01ED	2	2	2	35	64	01EE	18	18	18	35	65
01EF	2	2	2	14	27	01F0	38	38	38	38	38	01F1	39	39	38	38	38	01F2	40	40	38	38	38	01F3	17	17	17	27	38
01F4	58	58	58	58	58	01F5	2	2	58	58	58	01F6	40	40	40	40	40	01F7	2	2	2	27	27	01F8	3	3	3	3	3
01F9	39	39	39	39	39	01FA	17	17	3	3	3	01FB	17	17	17	17	17	01FC	35	35	35	35	35	01FD	2	2	2	35	35
01FE	18	18	18	18	18	01FF	2	2	2	2	2																			



(a) original



(b) (SYM-picture)$_2$

Fig.4-48 SYM-picture made by the transcription table with local noise elimination capability.

IV-3-3-2 An Automaton Description Which Checks the Constraint of Lines

To estimate the width of the digital lines we must group the symbols which are parts of the digital line. Since the unit mesh is a square, this grouping is considered to be of two kinds, horizontal and vertical directions as shown in Fig.4-49. We call this a "line element".

For detecting the line elements, we consider a finite states automaton, which checks the constraint of lines in the symbol sequences of SYM-pictures. Before describing this, another grouping of the set of legal patterns must be stated.

The legal patterns except W pattern (black legal patterns) are divided into three groups taking the structure in them into consideration. There are two directions, i.e. horizontal and vertical, but here we describe the horizontal direction as an example. The same is true in the vertical direction.

Among 65 legal patterns, those which can exist at the point of the constraint of the

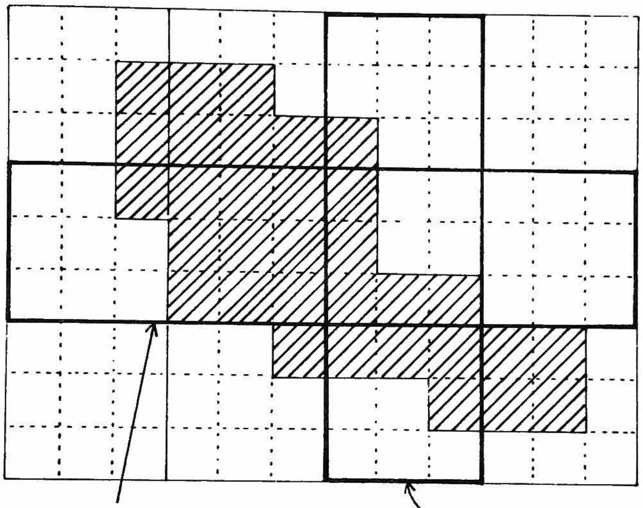


Fig.4-49 Line elements.

line in the right position of W pattern are classified into group B', and those in the left position are into group E'. From these sets, three sets, B (Begining legal patterns), M (Middle legal patterns), and E (Ending legal patterns) are defined as follows :

$$M = B' \cap E', \quad B = B' - M, \quad E = E' - M$$

It is clear that this is a partition of the set of black legal patterns. The numbers of elements to be included in B, M and E are 16, 33 and 16, respectively.

From the definition of the digital line, the groups B and E are still more divided. 1 group consists of the legal patterns which are the digital lines alone. 2 group consists of those which are only the gaps with sufficient width by themselves, and, others are 3 group. In the case of M, there do not exist white gaps, so that group 1 is the same, and group 2 is the same as group 3 in the previous case.

To summarize the discussion above, the legal patterns are divided into 9 sets, W, B1, B2, B3, M1, M2, E1, E2 and E3. Fig.4-50 shows the groups of each legal pattern, the upper is from the horizontal direction and the lower from the vertical direction.

Based on this grouping, we construct an automaton which checks the constraint of

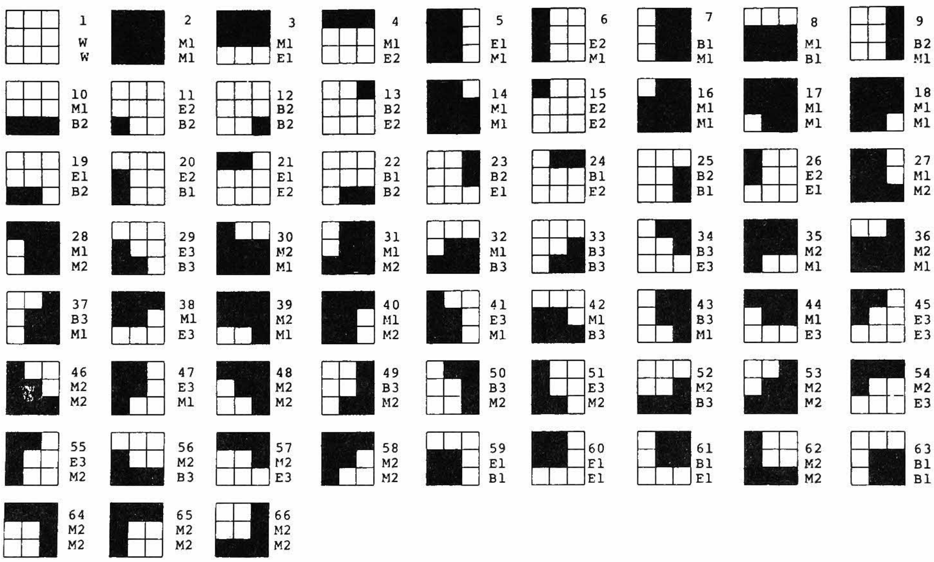


Fig.4-50 Groups of legal patterns for the automaton.

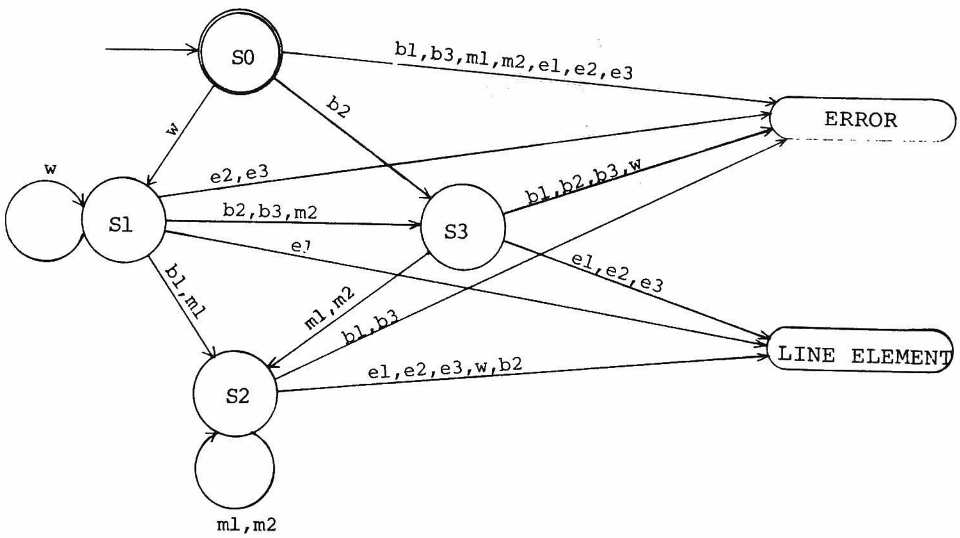


Fig.4-51 State transition diagram for the automaton.

lines and accepts the line elements. The notation here is that a small character represents one of the elements, for example, b1∈B1 etc. The state transition diagram is shown in Fig.4-51. The equivalent regular grammer is as follows:

non terminal symbol -- W_n : LNE, A, B, C

terminal symbol — W_t : b1, b2, b3, m1, m2, m3, e1, e2, e3, w

rewriting rules -- R : LNE→b2A, LNE→wB, A→m1C, A→m2C,
A→e1, A→e2, A→e3, B→b1C,
B→m1C, B→b3A, B→m2A, B→b2A,
B→e1, C→m1C, C→m2C, C→e1,
C→e2, C→e3, C→w, C→b2

This automaton acts on SYM-pictures and checks the symbol sequence in the continuous positions. This check is only a syntax one, and for some sequences even though this check is passed, they may not be digital lines. To the contrary, all the digital lines can pass this check. An example of the result of this automaton is shown in Fig.4-52. In this figure, the accepted sequences are denoted "LNE" and rejected are "ERROR". Therefore the symbol sequences decided to be "ERROR" are considered to be noise or some distorted signal.

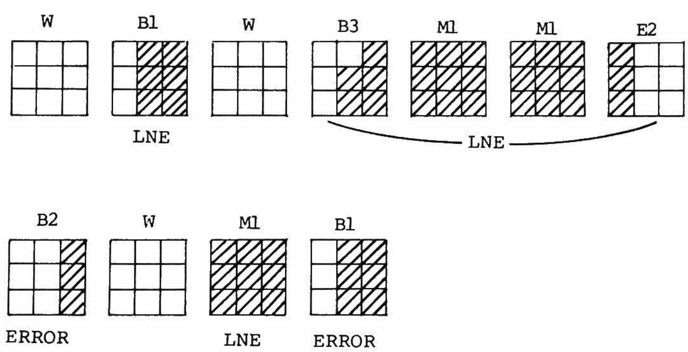


Fig.4-52 Examples of the acceptance sequence of the automaton.

IV-3-3-3 Estimation of the Line Width

The width of the line elements accepted by the automaton are calculated easily by taking the symbols in the sequences into consideration. The width is considered to be the sum of the black cells with the maximum length in the symbols of the accepted sequence. For example, the accepted sequences shown in Fig.4-52 have width 2, 9 and 3 from the left respectively.

If we assign this width to all the symbols in the accepted sequences in two directions, each symbol except W pattern in the SYM-picture has two values, that is, one corresponds to the width in the horizontal direction and the other in the vertical direction. The sequences to be rejected are not calculated at all.

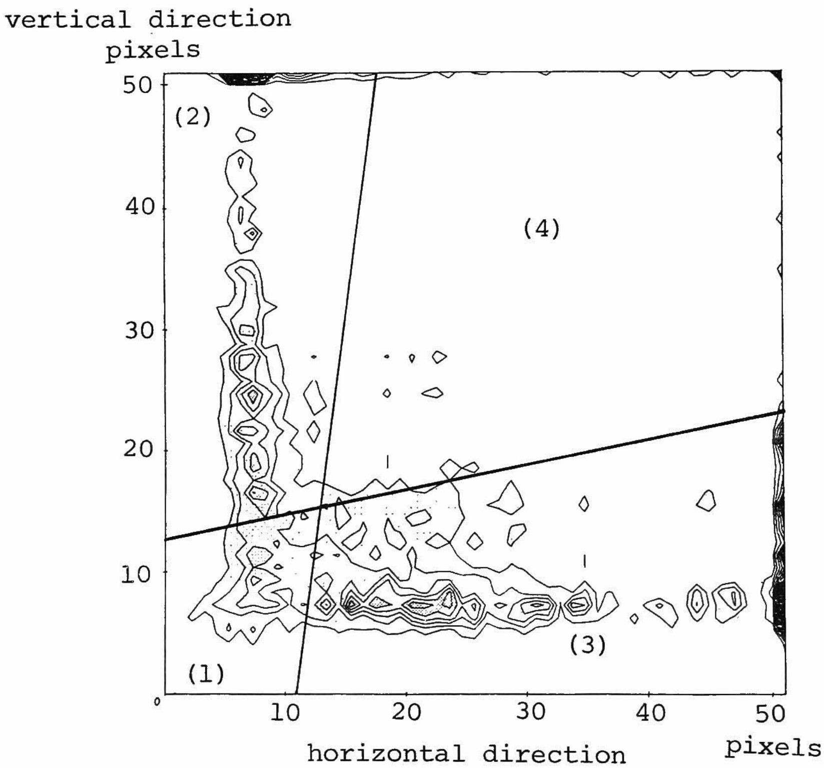


Fig.4-53 Histogram of line width in two dimensional space.

For CCITT's test chart NO.2, these two values for each symbol are calculated and plotted to make the histogram in two dimensions. This is shown in Fig.4-53. The horizontal axis corresponds to the width in the horizontal direction and the vertical one in the vertical direction. The contours represent the same populations. From this figure, we are able to divide the symbols in the digital lines into 4 classes according to their width.

(1) The digital lines with narrow widths in both directions are considered to be slant lines or digital points.

(2) The digital lines with relatively large width in the vertical direction and at the same time with relatively short width in the horizontal direction, correspond to vertical lines.

(3) With the terms "horizontal" and "vertical" in description (2) exchanged, digital lines are considered to be horizontal lines.

(4) The digital lines with relatively large widths in both directions correspond to crossing points of the vertical and the horizontal digital lines.

The thresholds for distinguishing them are decided from this histogram, intrinsically. Though to decide only the thresholds, it is better with two independent histograms not to use a representation in two dimensional space as in Fig.4-53, because of the configuration of the histogram. In practice, it is difficult to decide threshold values in this histogram, so that we estimate the width of the line from two independent ones, in the horizontal and vertical directions.

Fig.4-54 gives examples of the histograms of the line width in the horizontal direction. The data is cut out from CCITT's test chart NO.5, each of which is selected in three image quality classes. Fig.4-54(a) corresponds to "stained", (b) "good" and (c) "blurred" quality.

The first dominant peak is considered to correspond to the width of the vertical lines, but there are many slant lines in the data, so, here, the first dominant valley is considered to be the width of the digital lines. This estimation is good in the histograms of the "good quality", but it will be difficult in the histograms of the low-quality images.

To verify these estimations, we select the digital lines to be estimated as the vertical lines and the horizontal lines and show them in Fig.4-55 and Fig.4-56, respectively. The data are CCITTs' NO.2. From these figures, these estimations seem to be good.

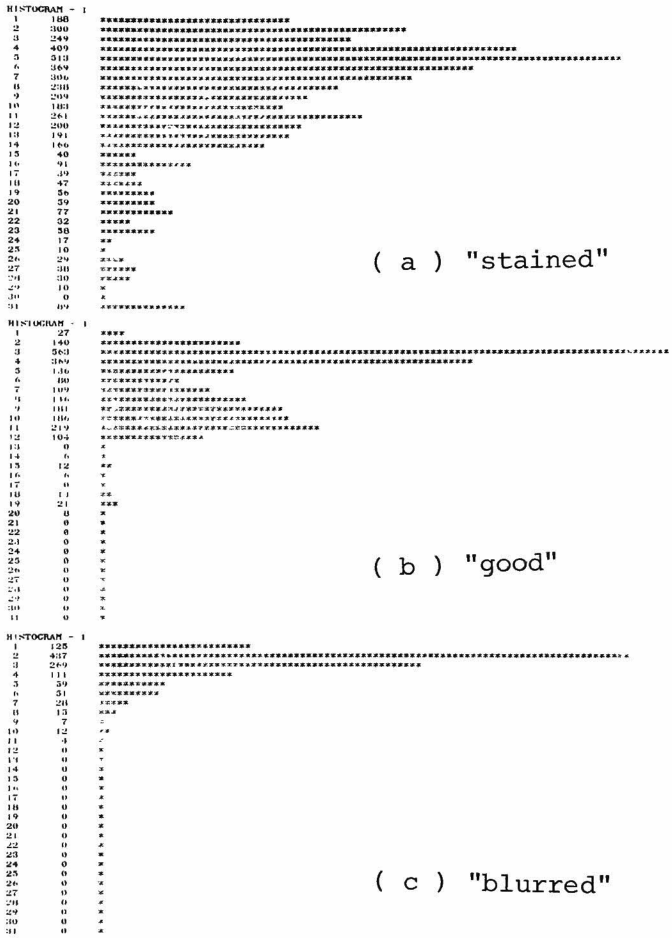


Fig.4-54 Histograms for three image qualities.

IV-3-3-4 Procedure of Noise Elimination

IV-3-3-4-1 Case of "Stained" Images

Fig.4-57(a) is an image with 255 × 255 size cut out from CCITT NO.5 data classified into the "stained" class. Fig.4-58(a) is the histogram of the width of the accepted line elements, and (b) is the histogram of the error symbols. One of the features in "stained"

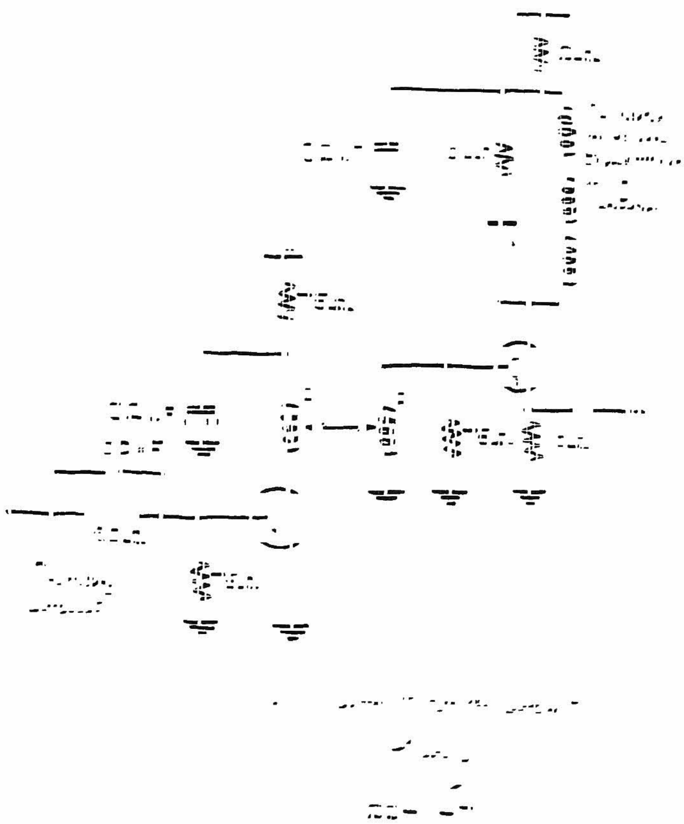


Fig.4-55 Horizontal lines to be estimated.

images is said to have much noise in the background. Generally, this noise is not accepted as a line element, so that from the histogram of the error symbols in Fig.4-58(b), the value of the first valley is selected as the threshold value T_e to distinguish signal from noise. As stated in the previous section, the value of the first dominant valley is estimated to be the line width and denoted T_w.

Based on these two estimated values, T_e and T_w, the process eliminates noise as follows.

(1) The symbols with either of the widths less than T_e are eliminated.

(2) The symbols with both of the widths less than T_w are eliminated, unless there does not exist any symbol with either of the widths greater than the T_w in its 4-neighbours.

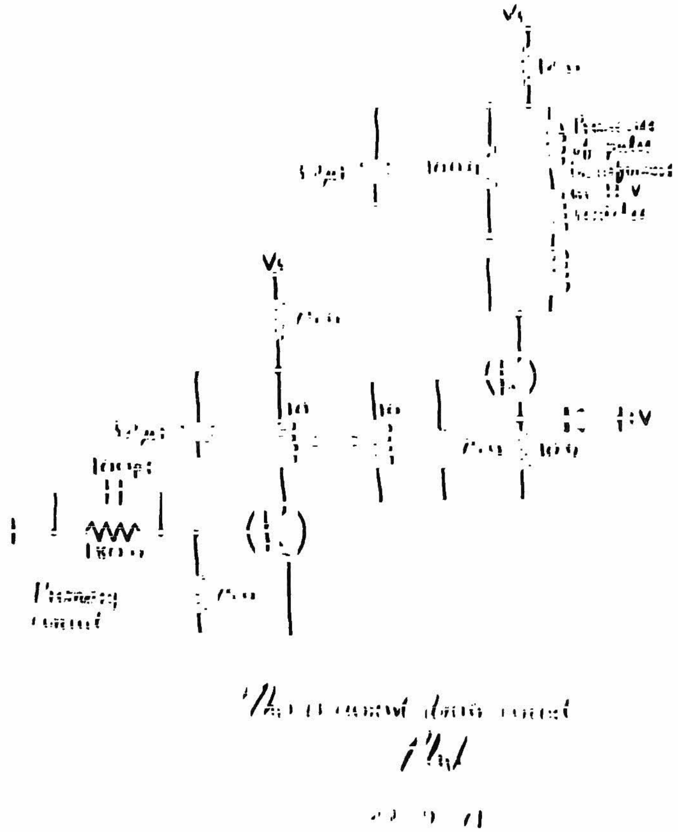
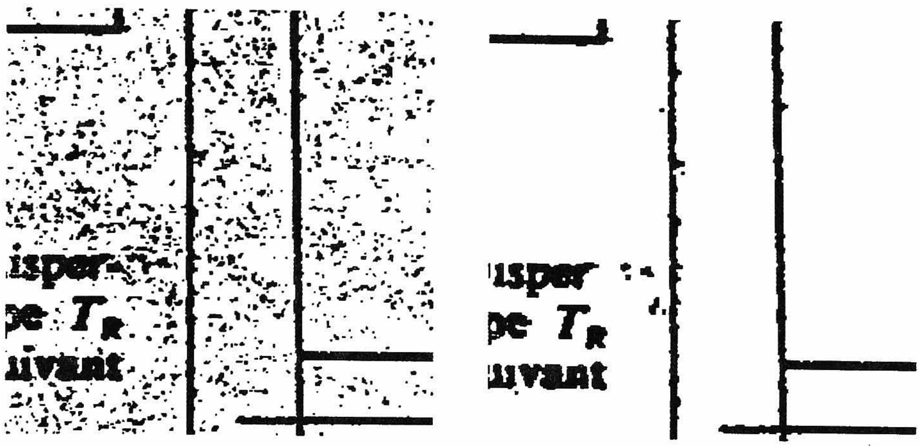


Fig.4-56 Vertical lines to be estimated.

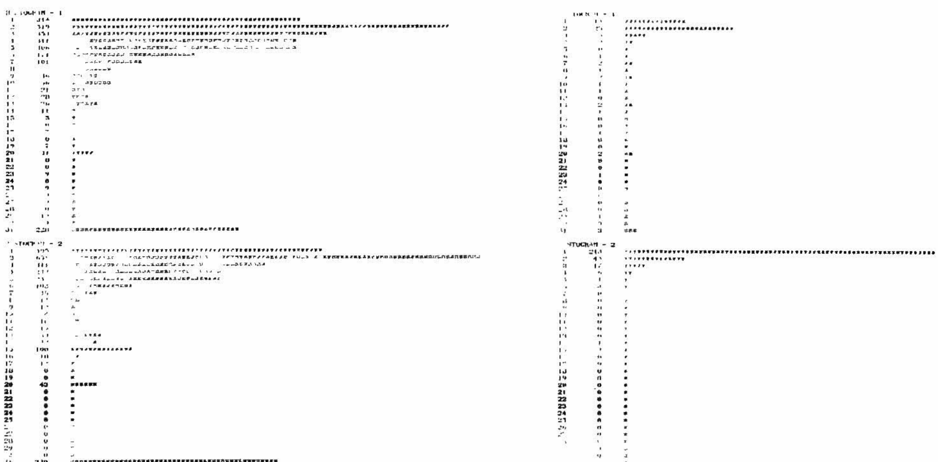
(3) These steps (1) and (2) can not eliminate the small noise around the digital lines. So the next step is to transform the resultant image to an LSC-picture.

The final result is shown in Fig4-57(b). This method is basically to distinguish noise from signal according to the size of their area, therefore, the noise with large area is not eliminated. This is the limit of this method. To eliminate them, a little more information than that of the digital line, i.e. some real world constraints is necessary.



(a) original (b) result

Fig.4-57 Experimental result for "stained" image.

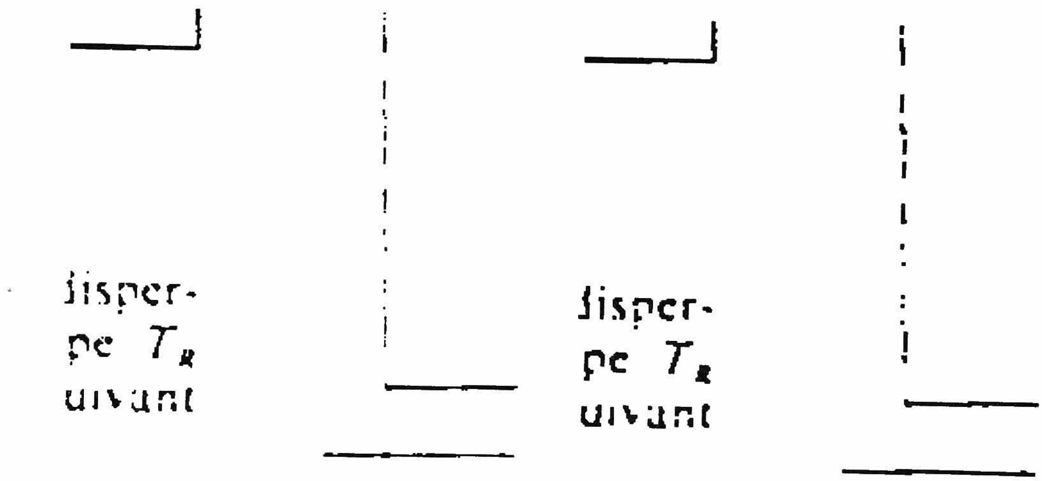


(a) accepted sequences (b) error sequences

Fig.4-58 Histograms of the image in Fig.4-57(a).

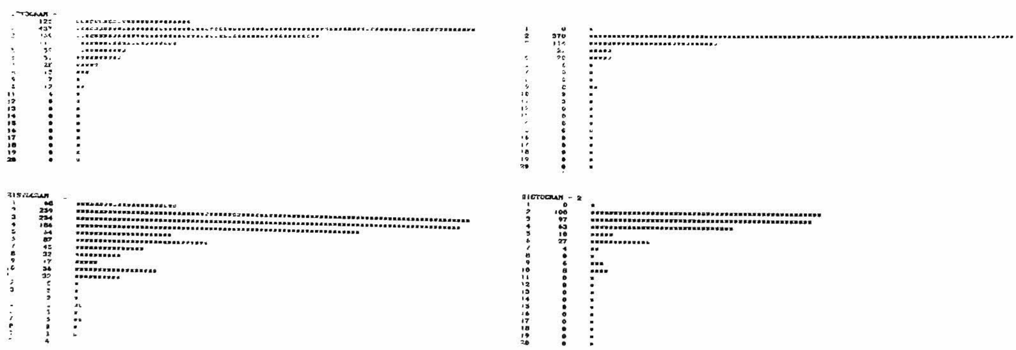
IV-3-3-4-2 Case of "Blurred" Images

Fig.4-59(a) is an image with 255 × 255 pixels from CCITT NO.5 classified into "blurred" class. As in the previous section, the histograms of the width of the accepted line elements and of the error symbols are shown in Fig.4-60(a) and (b), respectively. In



(a) original (b) result

Fig.4-59 Experimental result for "blurred" image.



(a) accepted sequences (b) error sequences

Fig.4-60 Histograms of the image in Fig.4-59(a).

this case, the lines are made thick.

The symbols satisfying the following conditions are replaced to make the digital lines thick.

(1) The begining symbol of one line element must belong to one of the 4-neighbours of the ending symbol of another line element.

(2) These two line elements have the same direction, i.e. the direction of the adjacent positions must be the same as the direction of both line elements, which are derived from the direction with larger width.

The histogram of the width of the line elements can not be used in this case, because the connection of the line elements largely depends on the semantics of the signal, i.e. some real world constraints. Finally, as in the case of the "stained" images, the resultant image is transformed to an LSC-picture. This image is shown in Fig.4-59(b). There are many gaps not filled. To do so, more information is necessary as in the case of "stained" images.

IV-3-4 Discussion

The adaptive system for noise elimination has been described in this section. The features of this system are as follows.

(1) The process depends on the image-quality of the input image, in this sense this system is adaptive. Even if the system cannot decide the image quality, the system works relatively well.

(2) The objective image size is 255 × 255 pixels, which atains to 1/60 of the whole image with size A4, so that for the partial degradation of the image quality, this system works fairly well, although there will be some problems around the boundaries.

(3) The actual processes work on the SYM-picture, which is a description of the original image. Therefore, it is possible to perform the processes faster and more intelligently.

(4) The system takes the statistics of the legal patterns, and gives the possibility of using them for the purposes of distinction of the characters, or division of the picture region and graph region etc.

Researches about the possibilities stated above are left for future study. The processes based on the 3×3 unit mesh have, we believe, many possibilities, and by extending this idea to multi-level images, great progress will be made in the field of document processing.

IV-4 Redundancy Reduction Coding

----- Application of MOLD Theory (2) -----

IV-4-1 The Problems

The standpoint of communication systems is to realize the sending data at the receiving side with high fidelity. In the field of image transmission, this objective also exists. It is well known that images have much redundancy, so that there has been much research into reducing this redundancy. Even if we restrict the range within the field of bi-level images, there have been many proposals based on various ideas. Recently, study group XIV of the CCITT has reported a Recomendation (T.4), which may be classified into one dimensional coding based on the run-length by using a 'modified Huffman code (MH code)', and into two dimensional coding known as a 'modified relative element address designate (Modified READ)' code.

Generally, the evaluation items for a coding method are:

(1) The compression ratio[1)]

(2) The influences of transmission errors

(3) Ease of realization as hardware (i.e. the complexity of the algorithm and the amount of the coding and decoding tables)

(4) Extensibility

These items are derived from the aspect of the digital transmission.

Our aspect is that the digital communication and information processing are intrinsically in the same field and we must devise a coding method which is available not only in the field of digital communication but of information processing as shown in Fig.4-61. From this standpoint, the coding methods which have been proposed appear not to be to our satisfaction. Almost all the coding methods are based on the signal, and the

1) Compression ratio is defined as the ratio of the number of the code sequences to the number of pixels for bi-level images.

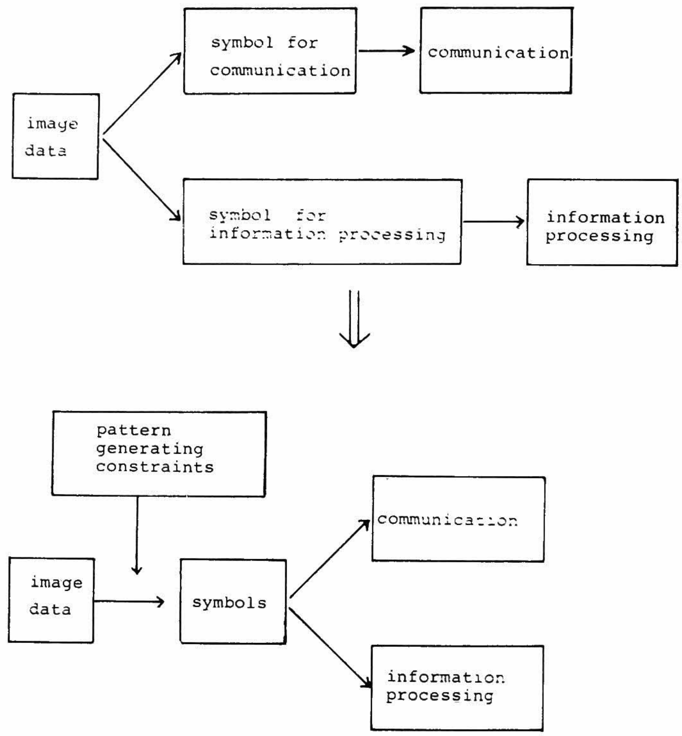


Fig.4-61 Strategy for selecting the coding symbols.

symbols for coding are selected only to reduce the redundancy.

The features of the coding from our aspect are as follows:

(1) We can observe the image only once in the decoding process, because the symbols for communication are also the same ones for those of information processing.

(2) We can correct transmission errors by using the redundancy, not only in the codewords themselves, but also in the relation of symbols.

(3) The coding process can have the ability of pre-processing the image i.e. noise elimination, to get the statistical data, etc..

(4) There will be the possibility of adaptive coding corresponding to the information region, that is, the character regions, graph regions, picture regions and so on.

In this section (IV-4), we describe the coding method in accordance with this concept. The symbols for coding are the legal patterns which are one class of representation of the pattern generating constraints, and are derived from the requirement of information processing. This coding method is evaluated from the standpoint of communication for the items described above.

IV-4-2 Standard Coding Techniques

There are excellent surveys [37,38,39,40] of the coding method of bi-level images. Therefore, we only describe the position of our coding method in several research areas.

The coding method can be classified into the following three categories according to the symbols for coding, i.e. (1) pixels (2) blocks and (3) run-length.

(1) Pixels
In this category the coding is done based on pixels. The representative methods are predictive coding (ordering algorithms) [41,43], growth geometry coding [42], or the coding for only transmitting boundary points [44,45].

(2) Blocks
To obtain a high compression ratio by simple methods, the coding symbols become the set of pixels (blocks). There are white block skipping (WBS) coding [46], adaptive block coding [47], and DF-expression [48] etc..

(3) Run-length
Run-length is the most popular symbol for coding. The standard coding method recommended by CCITT are classified in this category. Representatives are Modified Huffman coding and Modified READ coding [49,50].

Our coding method is classified into the second category i.e. blocks for coding symbols, and has the characteristic of approximate coding. There are a few coding methods developed from this standpoint. The DF-expression developed by Kawaguchi [48] is fairly good from the point of information processing and of compression ratio, but its ability in information processing is rather restricted, and coding and decoding algorithms are complex. The combined matching system developed by Pratt [51] has low ability for character identification, and this coding method divides the objective image into the character parts and the residue. By doing this, a high compression ratio is obtained, but the algorithm is complex and if the image contains noise, the efficiency will go down.

Before describing our coding methods in detail, we first, briefly, comment on the standard coding methods i.e. MH and MR [49,50] because ours take advantage of the concepts of these methods.

(1) Modified Huffman coding (one dimensional coding)

The coding strategy is based on the coding for each scan line. The last code of coding one scan line is EOL which has a unique bit sequence which does not appear in any combination of the code words. Each line consists of a sequence of alternating black and white runs, and is assumed to begin with white ones. This coding schema takes run-length set as the coding symbols, and for these symbols two types of code words are assigned. One is called the terminating code (TC) and has code words corresponding to run-length from 0 to 63. The other is called the make up code (MUC) which represents a run-length value of 64×N. If a run-length value is greater than 63, an MUC is transmitted followed by a TC. Otherwise, only a TC is transmitted.

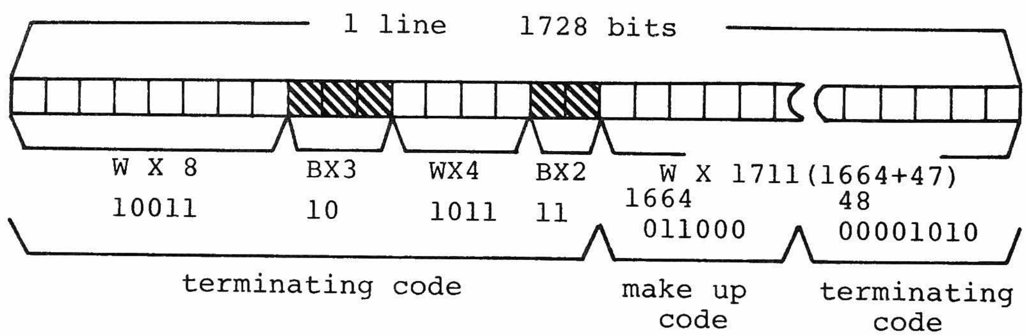


Fig.4-62 Examples of MH coding.

Fig.4-62 shows an example of Modified Huffman coding.

(2) Modified READ coding (two dimensional coding)

The modified READ coding is a line-by-line schema and the line to be coded is called "coding line" and the line which lies immediately above the coding line and just coded is called "reference line".

Before describing coding procedure, preparation is needed about the definition of the changing pixels and the three coding modes.

(i) Definition of changing pixels

A changing pixel is an element whose color is different from that of the previous pixel along the same line. There are five kinds of changing pixels.

a0: the reference or starting changing pixel on the coding line

a1: the next changing pixel to the right of a0 on the coding line

a2: the next changing pixel to the right of a1 on the coding line

b1: the next changing pixel on the reference line to the right of a0 having the same color as a1

b2: the next changing pixel to the right of b1 on the reference line

(ii) Definition of coding mode

The procedure utilizes three coding modes.

1) Pass mode

This mode is identified when the position of b2 lies to the left of a1

2) Vertical mode

This mode is identified when the position of a1 is coded relative to the position b1 and the relative distance is $|a1b1|<3$

3) Horizontal mode

This mode is the other case, and the run-lengths $|a0a1|$ and $|a1a2|$ on the coding line are coded at the same time

(iii) Coding procedure

The coding procedure is formally defined by the flow diagram shown in Fig.4-63.

(step 1)

If b2 is detected before a1, then 'pass mode' is identified.

The reference pixel a0 is set on the pixel below b2.(Fig.4-64(a))

(step 2)

Determine whether vertical mode or horizontal mode

i) If $|a1b1|<3$ then vertical mode. Set a0 on the position of a1 in preparation for next coding. (Fig.4-64(b))

ii) If $|a1b1|>3$ then horizontal mode. Set a2 on the position of a0. (See Fig.4-64(c))

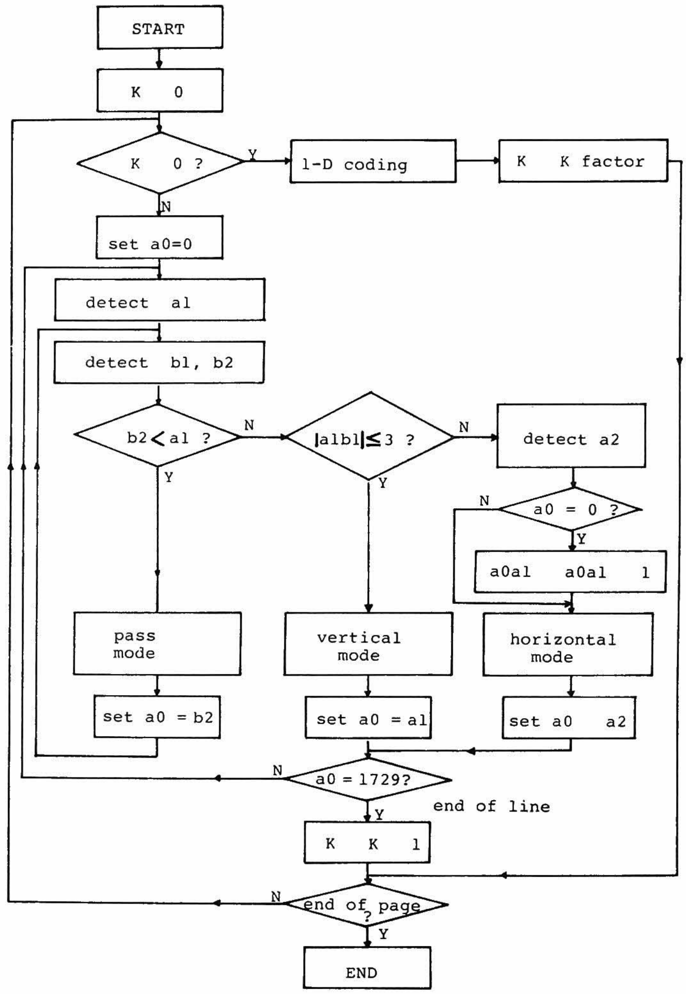
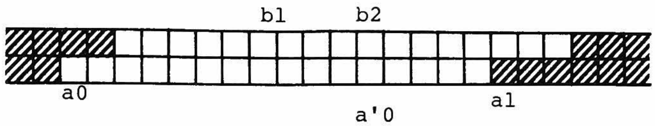


Fig.4-63 Coding procedure of MR coding.

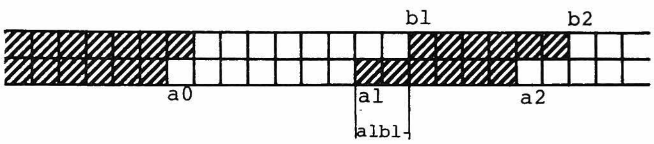
Fig.4-64(d) is an example of Modified READ coding.

The theoretical and practical compression ratios of these standard methods for CCITT's test charts are summarized in Table 4-11. From this table, the coding method seems to be very good, because the efficiency of the code is about 93%.

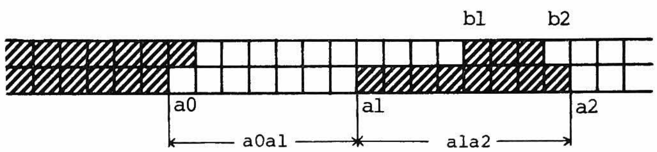
In our coding methods, the image for coding is an LSC-picture as defined in section IV-2. Their pixels are the legal patterns defined by 3×3 unit meshes and the relations in their 8-neighbours have been satisfied. It is this redundancy we want to take advantage of in our coding methods.



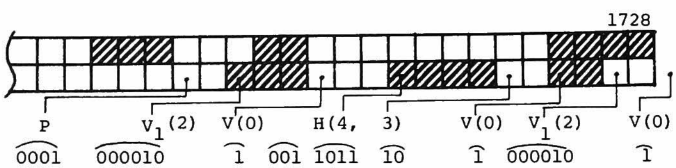
(a) pass mode



(b) vertical mode



(c) horizontal mode



(d) an example of MR coding

Fig.4-64 Examples of MR coding.

Table 4-11 Compression ratio for MH coding and MR coding.

	entropies		coding level	
	M H	M R	M H	M R
CCITT NO.1	20.64	33.49	15.61	22.06
CCITT NO.2	23.48	60.61	15.56	27.57
CCITT NO.3	11.46	22.68	9.16	14.48
CCITT NO.4	6.14	8.71	5.26	6.95
CCITT NO.5	10.66	19.76	8.76	13.35
CCITT NO.6	15.44	39.22	10.49	19.41
CCITT NO.7	7.89	8.59	5.16	6.83
CCITT NO.8	12.61	33.56	8.37	16.29

IV-4-3 Coding Method for LSC-pictures

IV-4-3-1 Zero Memory Coding (ZMC)

The simplest idea for coding LSC-pictures is to code each symbol one by one independently. The number of the symbols is 66. We investigated the occurrence probabilities for the symbols and assigned a code word to each symbol by the method of Huffman.

This coding method is considered to block 9 pixels (our coding symbol is defined in the unit mesh with size 3×3 pixels). Even in the most ideal case, each symbol occupies 1 bit code word. The compression ratio in this case is, at most, 9. We must enlarge the size of the unit mesh in order to make the compression ratio higher, which requires the increase of the number of legal patterns. This degrades the compression ratio. Therefore, we are obliged to introduce some technique named 'approximate coding', which somewhat grades down the image qualities.

There have been a few methods proposed on the basis of this idea.

Tominaga [35] introduced the 4×4, 8×8 and 16×16 blocks aiming at a high compression ratio. He calculates the statistics of the objective image and selects at most 50 symbols among the possible patterns for coding. When applied to weather charts, English and Japanese characters, he got the average compression ratios of 6.9, 8.2, and 8.2 for 4×4, 8×8, and 16×16 blocks, respectively.

Knudson [36] introduced 8×8 block and found that a subset of 62 patterns out of the possible 2^{64} was sufficient to obtain adequate image quality for practical use. He got a compression ratio of 10 for newspaper text and graphics without any coding method.

Zero memory coding method for LSC-pictures is almost the same as these methods. The main difference is that our method is an approximate coding in a strict sense but the image quality is not worse compared with that of the original. From the standpoint of the compression ratio, our method is not so good. From the code words in Appendix-C, it can be seen that the code efficiency will be rather low.

A prominent feature of ZMC is the possibility for error detection and correction by utilizing the relation among symbols. In an LSC-picture, a symbol is restricted by the symbols in its 8-neighbours. ZMC does not take advantage of this redundancy, so that this property is available for error detection and correction. On the other hand, it may be possible to use this property to obtain a higher compression ratio. Other coding methods for LSC-pictures described in this section are from this standpoint.

IV-4-3-2 One Dimensional Coding (ODC)

In this section we introduce a run-length coding method like the 'Modified Huffman' described in section IV-4-2. The run-length coding is only suitable for bi-level images. Each pixel in an LSC-picture has values from 1 to 66 according to the kind of legal patterns. From these facts, we intend to extend the run-length coding method as follows.

(1) Only W pattern which consists of all white pixels is considered to be the white pixels in the run-length coding. The run-length means the number of continuous W patterns in the main scanning direction.

(2) The other 65 patterns (hereafter in this section called "black legal patterns") are considered to be the black pixels. The run-length of black patterns usually consists of various kinds of black patterns, so that it is not sufficient to transmit only the run-length number of black patterns. The symbols have a structural relation among their 8-neighbours, so that this redundancy may be utilized.

Some considerations are necessary to compare the ODC with the coding on the pixel level. The following are assumed.

* Statistical property

the occurrence probabilities of run-length

for white runs $0 \le P_w(k) \le 1$

for black runs $0 \le P_b(m) \le 1$

where m, k : the length of each run $0 \le m,k \le 1728$

the occurrence probability of the black legal patterns

$0<P_l(n)<1$ where n : the identifier of the black legal patterns $2 \le n \le 66$

Based on these assumptions, we calculate the average entropies for the one symbol line, i.e. 3 pixel lines as shown in Fig.4-65.

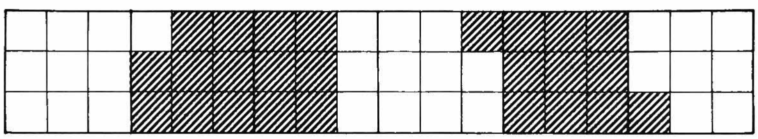


Fig.4-65 Example of one symbol line.

(1) For the white runs

* the pixel level

The average entropy of a white run-length is $-\sum_k P_w(k) \log P_w(k)$

We have three lines so that the sum is $-3\sum_k P_w(k) \log P_w(k)$

* the symbol level

We code three lines together. For any integer l, $1 \le l \le 576$

$P_w(3l)=P_w(3l-1)=P_w(3l-2) \triangleq P'_w(k)$ is assumed.

The average entropy of white symbol run-length is nearly equal to

$$-\sum_{l} P'_w(l) \log P'_w(l) = \sum_{k} P_w (k) \log P_w (k) - \log 3$$

(2) For the black runs

* the pixel level

As in the case of the white, the average entropy of three black runs is $-3\sum_{m} P_b (m) \log P_b (m)$

* the symbol level

The black runs of the three lines occupy $\lceil m'/3 \rceil$ symbols in the shortest case and $\lfloor (m' +1)/3 \rfloor +1$ symbols in the longest case. Where m' is the longest run in three black runs and $\lfloor x \rfloor$ means the maximum integer below x, and $\lceil y \rceil$ the minimum integer greater than y. Approximately we obtain the average entropy

$$\sum_{m} P_b (m) (\lfloor (m+1)/3 \rfloor +1) (-\sum_{n} P_l(n) \log P_l(n)) / 3$$

Therefore, we get the difference of the average entropies between the pixel level and the symbol level as follows:

$$-2\sum_{k} P_w (k) \log P_w (k) + \log 3 + \sum_{m} P_b (m) \{ 3 \log P_b (m) + (\lfloor (m+1)/3 \rfloor + 1) (-\sum_{n} P_l (n) \log P_l (n)) \} \quad \ldots\ldots \quad (4\text{-}1)$$

If this expression (4-1) is positive, the entropy on the pixel level is greater than that on the symbol level which is favorable to our coding. Otherwise the contrary is established. The first two terms are independent of m, i.e. corresponding to the width of digital lines. The last term depends on m, and as the average of m grows larger, it goes negative, and the expression (4-1) itself turns to be negative. The average value of m which makes the expression (4-1) nearly zero cannot be decided exactly because it depends mainly on the statistics of the image itself. Generally, ODC is suitable for a relatively small width of the digital lines, i.e. it is most efficient for approximately the same size of unit mesh and width of line.

To compress the redundancy, we introduce the simple Markov process for the coding of the black legal patterns. As stated in section IV-2, the kinds of the legal connections between two adjacent symbols are 1,582 of all possible number 4,356 (=2^{66}).

This constraint seems to have a contribution to the compression ratio. Indeed we can save this constraint for error detecting or correcting abilities as for ZMC.

One of the important features of this method, like ZMC, is the capability of doing the decoding process and the pre-processing (getting statistics, estimating the line width, deciding the characters region etc.) at the same time. This comes from the fact that the symbols for coding are selected in favor of the image processing, and the black legal patterns are coded one by one. This favorable feature will be proved in the near future.

IV-4-3-3 Two Dimensional Coding (TDC)

In this section, we try to apply the Modified READ coding technique described in section IV-4-2 to LSC-pictures. The black legal patterns are considered to be the black pixels and the W patterns are the white pixels as in the previous section.

(1) Definition of changing symbols

a0, a1, a2, b1 and b2 must all be the same as in the pixel level, though each pixel corresponds to a symbol.

(2) Definition of coding mode

(i) pass mode --- Same as in the pixel level

(ii) vertical mode --- The symbol itself consists of 3×3 pixels, so that the distance $|a1b1| \leq 1$ is sufficient to identify this mode. It corresponds to the range from three to five pixels.

(iii) horizontal mode --- If the distance $|a1b1| > 1$, then the run-length of $|a0a1|$ and $|a1a2|$ are coded. One of these must be the black legal patterns. They are coded on the simple Markov model like ODC.

(3) Coding procedure

The main part is shown in Fig.4-66.

(step1)

If b2 is detected before a1, then pass mode is identified. The reference symbol a0 is set on the symbol below b2. This step is exactly the same as that in the pixel level.

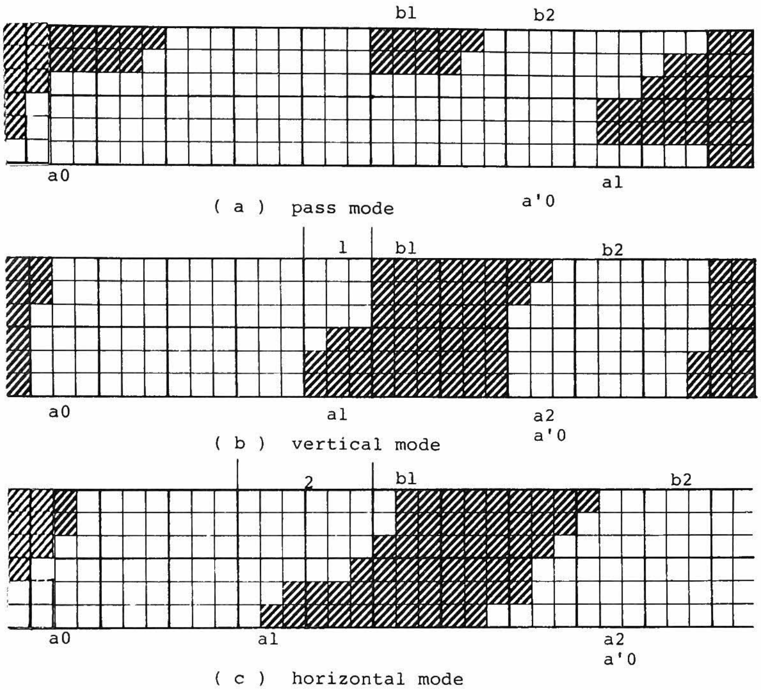


Fig.4-66 Coding procedure of TDC.

(step2)

Decide vertical mode or horizontal mode

(i) If $|a1b1| \leq 1$, then vertical mode. We must code $|a1a2|$ by the simple Markov model as well as $|a1b1|$. Next a0 is set on the position a2. This case is different from that in the pixel level.

(ii) If $|a1b1| \leq 1$, then horizontal mode is identified. Set a2 on the position of a0.

The main diference between MR and TDC is the point that the changing symbol a0 must be on the W patterns because TDC always does coding the black legal patterns and succeeding white patterns together. This means even in the vertical mode we must code the black legal patterns located from a1 to a2 as in the horizontal mode. MR coding includes some features of the method of specifying the changing pixels, because for bi-level images, the information about the position of the changing pixels is critical for compression ratio. However, in the case of the coding for LSC-pictures, the information about the position is not critical at all, but the information about the black legal patterns is important for compression ratio. Therefore high compression ratio is not expected as in the case of the pixel level. Still worse, the transmission errors seem to affect heavily the image quality due to the two dimensional coding.

IV-4-4 Experimental Results

IV-4-4-1 Simulation System of the Coding

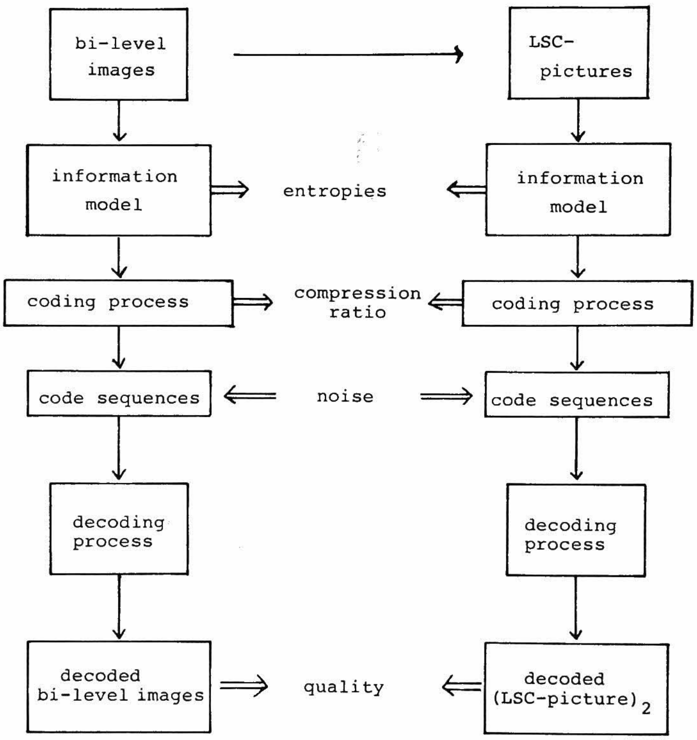


Fig.4-67 Simulation system.

The experiments have been done in the following steps. (See Fig.4-67)

(1) The models for information sources i.e. the images have been selected. Our purpose is to show the efficiency of our codings, ZMC, ODC and TDC, in comparison with the

Table 4-12 Compression ratio for ZMC, ODC and TDC.

	entropies			coding level			efficiency		
	ZMC	ODC	TDC	ZMC	ODC	TDC	ZMC	ODC	TDC
CCITT NO.1	15.67	28.26	32.26	6.93	27.32	28.37	0.44	0.97	0.88
CCITT NO.2	18.83	34.93	45.89	7.28	28.98	31.94	0.39	0.83	0.70
CCITT NO.3	10.36	17.04	20.17	5.93	16.38	17.18	0.57	0.96	0.85
CCITT NO.4	5.35	8.13	8.79	4.22	8.11	7.95	0.79	0.99	0.90
CCITT NO.5	9.98	15.73	18.25	5.84	15.41	15.95	0.59	0.98	0.87
CCITT NO.6	14.90	24.32	31.61	6.75	21.16	23.84	0.45	0.87	0.75
CCITT NO.8	6.81	16.85	22.68	4.26	7.40	7.66	0.63	0.44	0.34

standard coding techniques, MH and MR. The entropies of these models for the same information sources (here, we select the data of CCITT NO.1 to NO.8 except NO.7)[1] are investigated and summarized in Table 4-12.

(2) The coding is done for these models. The complete code words for our coding models, i.e. ZMC, ODC and TDC are listed in Appendix-C. The compression ratios of these models in the code sequence level including the EOL code word are investigated for the same data, and also shown in Table 4-12.

(3) For these code sequences, noise is superposed. The kinds of noise are random noise and burst noise.

(4) For each model the decoding processes are applied to the code sequence with the noise. If some errors are detected, the line is aborted and the previous line is repeated. Some examples of a part of the decoded image with errors are shown in Fig.4-68. The superposed noise is common in all images, and it is the random noise. The error ratio is settled at 10^{-4} for comparison.

1) NO7 does not satisfy the conditions for making LSC-pictures

In facsimile a photocell is caused to perform a raster scan over the subject copy. The variations of print density on the document cause the photocell to generate an analogous electrical video signal. This signal is used to modulate a carrier, which is transmitted to a romoto doctination ovor a radio or cable communications link.

At the remote terminal, demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

MH coding

In facsimile a photocell is caused to perform a raster scan ovor [illegible] ~~cause the photocell to generate an analogous~~ electrical video signal. This signal is used to modulate a carrier, which is transmitted to a ~~remote destination over a radio or cable communications link.~~

At the ~~remote terminal,~~ demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

MR coding

At the remote terminal, demodulation reconstructs the video signal, which io uocd to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

Probably you have uses for this facility in your organisation.

Yours sincerely,

ZMC

In facsimile a photocell is caused to perform a raster scan over the subject copy. The variations of print density on the document cause the photocell to generate an analogous electrical video signal. This signal is used to modulate a carrier, which is transmitted to a remote destination over a radio or cable communications link.

At the remote terminal, demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

ODC

In facsimile a photocell is caused to perform a raster scan over the subject copy. The variations of print density on the document cause the photocell to generate an analogous electrical video signal. This signal is used to modulate a carrier, which is transmitted to a remote destination over a radio or cable communications link

At the remote terminal, demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

TDC

Fig.4-68 Decoded images with errors.

IV-4-4-2 Evaluation of the Coding Schema

The results presented above in Table 4-12 and in Fig.4-68 are evaluated from the points described in section IV-4-1 in comparison with the standard coding methods. The numbers correspond to those in section IV-4-1.

(1) Compression ratio

From Table 4-12, the compression ratios of ODC and TDC are a little higher than that of MR in this data. This is because, even in ODC, the correlation between the lines is utilized owing to the symbols with the 3X3 unit meshes. As stated in section IV-4-3, however, as the width of the digital lines grows larger, the compression ratio decreases gradually.

The compression ratio of ZMC is very low compared with others, but ZMC has other advantages that will be described later. It is worth while saying that the compression ratio and error reduction capability are contradictory requirements.

(2) The influences of transmission errors

From Fig.4-68, the image of MH has the best capability against transmission errors. The codings for LSC-pictures have a disadvantage due to the 3×3 unit meshes. In other words, we code three scan lines all together, so if transmission errors occurred, at least three lines are not decoded correctly. Still worse, the disposal of the error suffered lines is not fit for our coding methods. Generally, correlations of every three lines are not so large. Therefore, the process to substitute the present line with the previous line makes the decoded image worse. We must select an other process for the error-suffered lines, but a great improvement is not expected. Therefore, some schemata of error correcting are inevitable in our coding methods and these are mentioned in the following works.

(3) The amount of hardware system

The coding and decoding algorithm of TDC is not as complex compared with that of MR. In the case of ODC, it is a little more complex than that of MH. Still, ZMC is the simplest coding algorithm.

The size of code table necessary is relatively large in ODC and TDC, i.e. about 1,650 code words, while about 100 code words are in MH and MR. In ZMC, there needs to be only 67 code words. The size of code tables for ODC and TDC may be practical considering recent development of LSI technology. If we want to decrease the sizes, we must give up using the simple Markov model for coding the black legal patterns. Instead, if coding is done on a memory-less model, the size of the coding and decoding table is at most only 150 entries. This may cause compression ratios to decrease.

The buffer necessary for transmission is not the problem. Strictly speaking, MH needs one scan line buffer, MR two scan lines, ZMC and ODC three scan lines, and TDC needs six scan lines.

(4) Extensibility

When the resolution is changed, or the paper size becomes larger, the degree of adaptability of these methods is very interesting. ZMC is applied as it is, but the other coding methods must have additions to the code tables because they are dependent on the run-length coding. In fact, CCITT's recommendation includes the extention code up to 2,048, which corresponds to B4 size with the fine detail mode. Other changes do not effect these methods as much.

These items are common in the field of communication. We want to add another item, the affinity of the information processing. From this standpoint, MR is the worst of all. The symbols for coding are either the position of the changing pixels or the run-length between them. Therefore, we cannot process at all. The coding symbols for MH are the run-length, and we can perform some information processing based on it.

The coding methods based on legal patterns are very well fitted for the image processing, because traditionally the 3×3 window has been used, and most of the processing algorithms are at least two scans of the image, once for observation and the second for processing. One of the features of these codings is that we can decode and observe the image at the same time. From the simple observation, for example, we can decide the image quality as described in section IV-3, and there will probably exist other useful pre-processing.

IV-4-5 Discussion

In this section IV-4, we described the coding schema for LSC-pictures. The advantages of these methods are summarized as follows.

(1) The compression ratio is almost as high as the standard ones, MH and MR.

(2) The influences of the transmission errors are worse than those of the standard ones unless some error correcting methods are introduced.

(3) The tables for coding and decoding are much bigger than those of MH and MR. Giving up the Markov model for coding the black legal patterns, they are almost the same size as those of MH and MR.

(4) The affinity of the information processing is much better than the others. This becomes, we believe, the most important property in future communication.

To restrict discussion to our three coding methods, ZMC is the simplest and has many advantages, but its compression ratio is the worst of all. This is relatively an important item for communication, therefore ZMC is not said to be a good coding.

TDC has the highest compression ratio of all, but the coding and decoding algorithm is rather complex, and the influences of transmission errors affect it significantly. The improvement of compression ratio in comparison with ODC is not worth these disadvantages.

ODC seems to be the best coding of all. The compression ratio is relatively high, the algorithm is rather simple and the influences of errors affect only one line. The symbols for coding are defined intrinsically by 3×3 unit meshes, so that the correlation on the vertical direction is also utilized in ODC of horizontal direction. The problems in deciding the coding method in general must be considered among the items of the compression ratio, the complexity of the coding/decoding algorithms and the influences of errors. From this viewpoint, it seems to be needless to use TDC for LSC-pictures unless it uses the correlation among the black legal patterns, not only in the horizontal, but also in the vertical direction. This is because the two dimensional prediction model is able to greatly compress redundancy, as shown in Fig.4-13.

Table 4-13 Entropies of CCITTs' test charts as to various information source models.

	legal patterns		
	zero memory	simple markov	2-D prediction
CCITT NO.1	15.67	27.47	56.82
CCITT NO.2	18.83	42.74	78.68
CCITT NO.3	10.36	17.76	40.00
CCITT NO.4	5.35	8.07	15.11
CCITT NO.5	9.98	16.23	35.46
CCITT NO.6	14.90	27.17	66.53
CCITT NO.8	6.81	21.46	48.78

Digital communication will become more closely related to information porcessing in some senses. In this case, the coding symbols are more similar to those of the information processing as is described in this section. There are left many problems of other methods for LSC-pictures, but we hope this section may serve to show the direction to proceed in future.

IV-5 Reconstruction of Hand Sketched Line Drawings

----- Application of MOLD Theory (3) -----

IV-5-1 The Problems

In our terminology, line drawings consist of character regions and diagrams (see Fig.4-69). Diagrams are difficult to process for coding by computers, because the lines included have information about the positions, curvatures and widths.

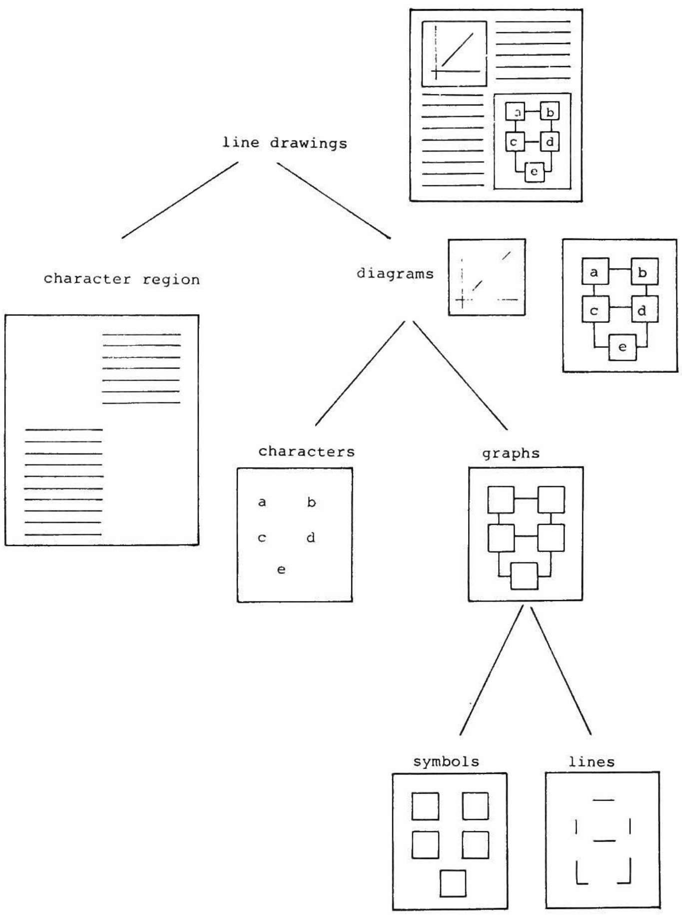


Fig.4-69 Terminology of this section IV-5.

The development of computers and I/O devices make it necessary to process these diagrams. In our society, there are many kinds of diagrams and the problems to store, retrieve, and update them, remain in future [60,61]. This field is basically the same as that of CAD or computer graphics, where the manual input process has been very hard and tiresome work.

From the viewpoint of processing diagrams, they may be divided into two categories as below.

(1) The lines themselves included in the diagram have a significant meaning; an important piece of information is, then, positions and curvatures. Examples in this category are draftings.

(2) The graphical symbols representing some meanings are already defined. The key information of this kind of diagram is the topological relations among symbols i.e. the connectivity of symbols. Logical diagrams, block diagrams, circuit diagrams and flow charts are examples of this category.

The diagrams in the first category must be drawn accurately. Our objective diagrams, therefore, are in the second category, and we can derive the key information from them even if they are hand sketched diagrams.

In this section, we discuss a method for the input process of hand sketched diagrams to free people from this tiresome work. Generally, this process is considered to be the process of transforming dot data into vector form. The simple idea is to do this on an on-line system [62].

There are two methods worth commenting upon. Ramachandran [52] developed a method named "vector coding". For the purpose of information compression, he transformed the dot data to vector form. This method was based on the run-length process, so that only one scanning was necessary, but a large problem existed in the process for nearly horizontal lines which occurred frequently in the diagrams in the second category.

Yoshida <u>et</u> <u>al</u> [59] handled hand sketched data written on section papers (meshes). The diagrams were restricted to those of printed patterns. According to the section of the paper, the symbols were recognized and the lines were transformed to vector forms. They also made a hardware system. Many restrictions existed in writing diagrams.

Our purpose is only to extract the topological information and reconstruct the hand sketched diagrams. If we want to use the extracted data to be equivalent to the field of computer graphics, the recognition of the symbols is essential. We take advantage of SYM-pictures to make the process faster and at a higher processing level, because the symbols already represent the local features and the automaton described in section IV-3-3 is available.

First, a method to distinguish between characters and graphs is described in the next section. Only the regions of the graph are selected for further processing. Also, for these graph regions, the feature points necessary to extract topological representations are difined and extracted based on the results of the automaton (see section IV-5-3). For these feature points the information about the connections, which feature points are connected to others, is investigated and it is determined whether they are curves or straight lines (IV-5-4). The topological descriptions are constructed from this information, and the reconstructions are done. These are described in section IV-5-5.

IV-5-2 Distinction between Characters and Graphs

Diagrams generally contain characters. The properties of the characters are very different from those of graphs, and the probelms often arise that the characters disturb the process of handling the graphs. There is an important concept 'that we must classify the input data in the case that there are various kinds of regions'. Therefore, first, we consider the method for distinguishing characters from graphs.

The diagrams in the second category usually consist of three components: lines, symbols and characters. In order of complexity, the characters are first, the symbols are second and the lines are last. To make the method a general one, knowledge about the input diagrams should not be introduced at an early stage. For our purpose, the symbols are considered to be on the same level as the lines and these two are called graphs. If recognition of these symbols is necessary, we may be able to do it on the basis of our final description.

The key property for distinguishing characters and graphs exists in the characters and in the graphs. For example of the former, the character regions are compact, or the density of the black pixels is high around the characters etc. Here, we comment on two of the proposed methods.

Terajima et al [54] developed a method based on compactness of the character regions with the connection information on the black pixels. They settled the restriction of the size of the characters, and of the gaps within the characters and between the graphs and the characters. These restrictions seems to be very strong.

Iwaki et al [55] developed a method based on the density of the black pixels. They did not use any information about the connection. By calculation of the density of all the white pixels, the character regions were decided. There remain the problems of noise and of the large amount of calculation needed.

In contrast to these methods, we would like to utilize a property of the graphic region. The property we use here is that the graphic regions generally contain relatively long lines which lie nearly in the vertical or horizontal direction. The diagrams have great variety by themselves, so that there are no rules without exceptions. To restrict our discussion to the diagrams in the second category, the restriction derived from this property seems to be weaker than the others presented.

Explicitly, our restrictions for the input diagrams are:

(1) The characters must not touch the graphs.

(2) One connected graph region must include at least one relatively long horizontal or vertical line.

This leaves the problems concerning broken lines and separate symbols because of the connectivity.

Now, we explain the actual process. The steps of the process are as follows:

(step 1) The extraction of the horizontal or vertical lines

For a SYM-picture there exists an automaton to accept digital lines, and the widths of the line elements in the acceptance sequences are accumulated separately in the form of histograms. They are used to estimate the width of the digital lines. The estimated values are denoted V_w and H_w corresponding to the vertical and horizontal direction, respectively.

Emprically, other thresholds for the lower limit of the digital line length are defined as, four times as much as V_w and H_w, i.e. $V_l = 4 \times V_w$ and $H_l = 4 \times H_w$, respectively. The criteria decide the vertical lines or horizontal ones as follows:

(i) Vertical lines

The line elements with length in vertical direction more than V_l whose widths are less than H_w for a length of at least V_l.

(ii) Horizontal lines

The line elements with length in horizontal direction more than H_l whose widths are less than V_w for a length of at least H_l.

By examining the results of the automaton, the line elements which satisfy one of the above two criteria are extracted straightforwardly.

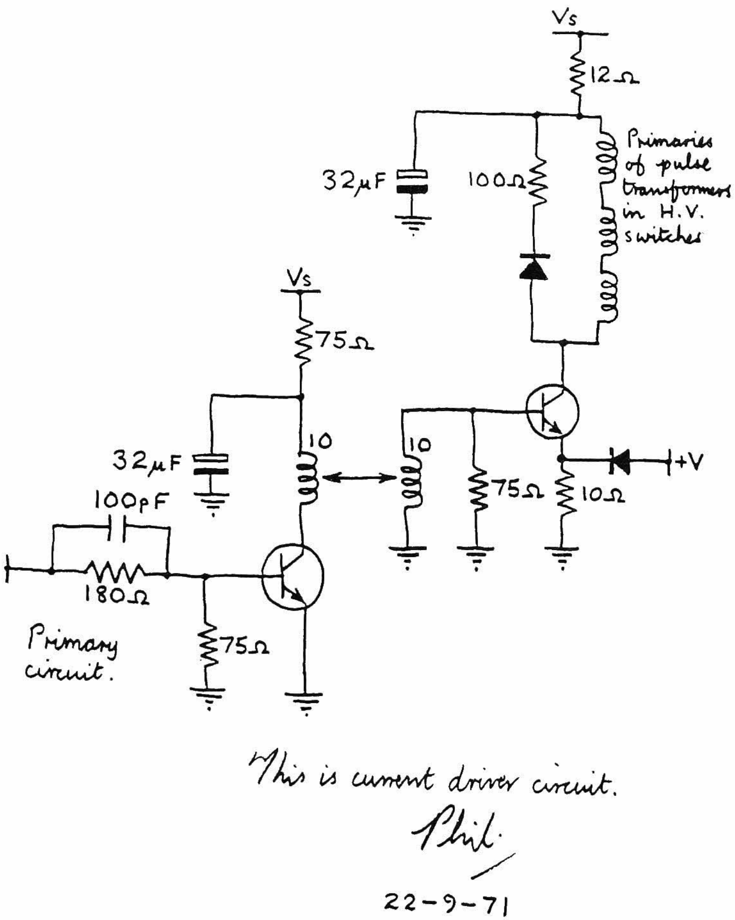
(step 2) The extraction of the connected regions

The process applied here is an iterative one. First, the parts extracted in step 1 are deleted. Then, for the remaining image, the automaton is applied again and the elements with width changing in comparison with the previous results are deleted. This process is repeated until no elements are deleted. The residue is the region of the characters.

The results for the data of CCITT NO.2 and NO.5 are shown in Fig.4-70 and Fig.4-71, respectively. In these two figures, (a) corresponds to the original data and (b) the character region. As feared, the symbols which represent condensers and earth are classified as characters, as shown in Fig.4-70, and the slant lines and the broken ones are also distinguished wrongly.

Because of the iteration, the process is rather simple but it takes much time. The programs to do this algorithm are written in PL/I and the computer used is M200. The time necessary for the process largely depends on the data. It takes about two minutes to process the data shown in Figs.4-70 and 4-71. We can easily save most time by removing the limitation of the memory size.

The errors will be corrected by introducing other properties to work together with this one for the slant lines. However, separate symbols and broken lines are intrinsic problems, therefore a higher level of knowledge about the diagrams should be introduced to distinguish them correctly.



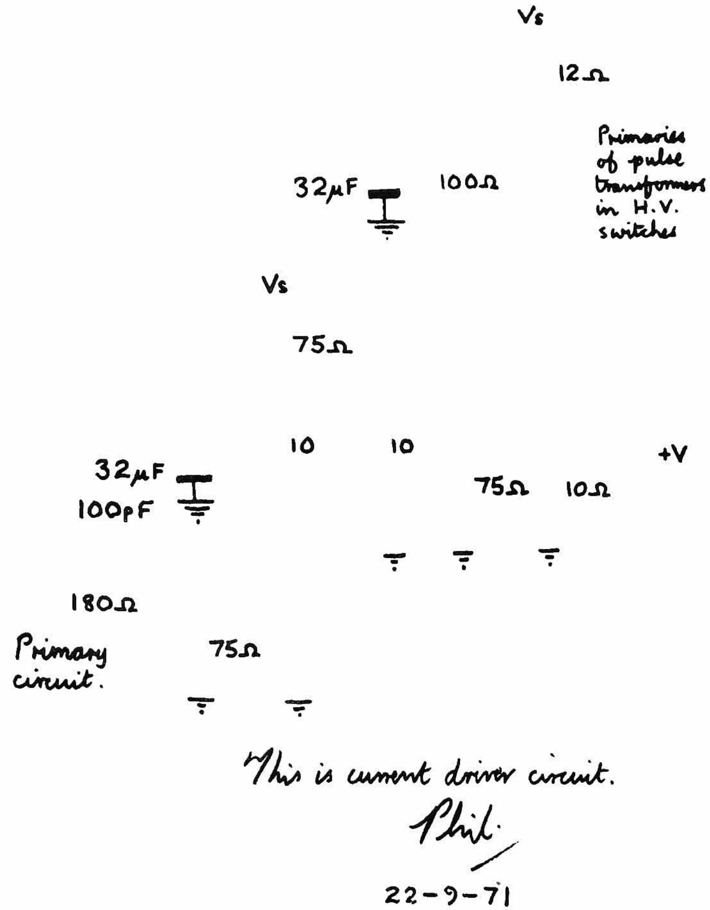
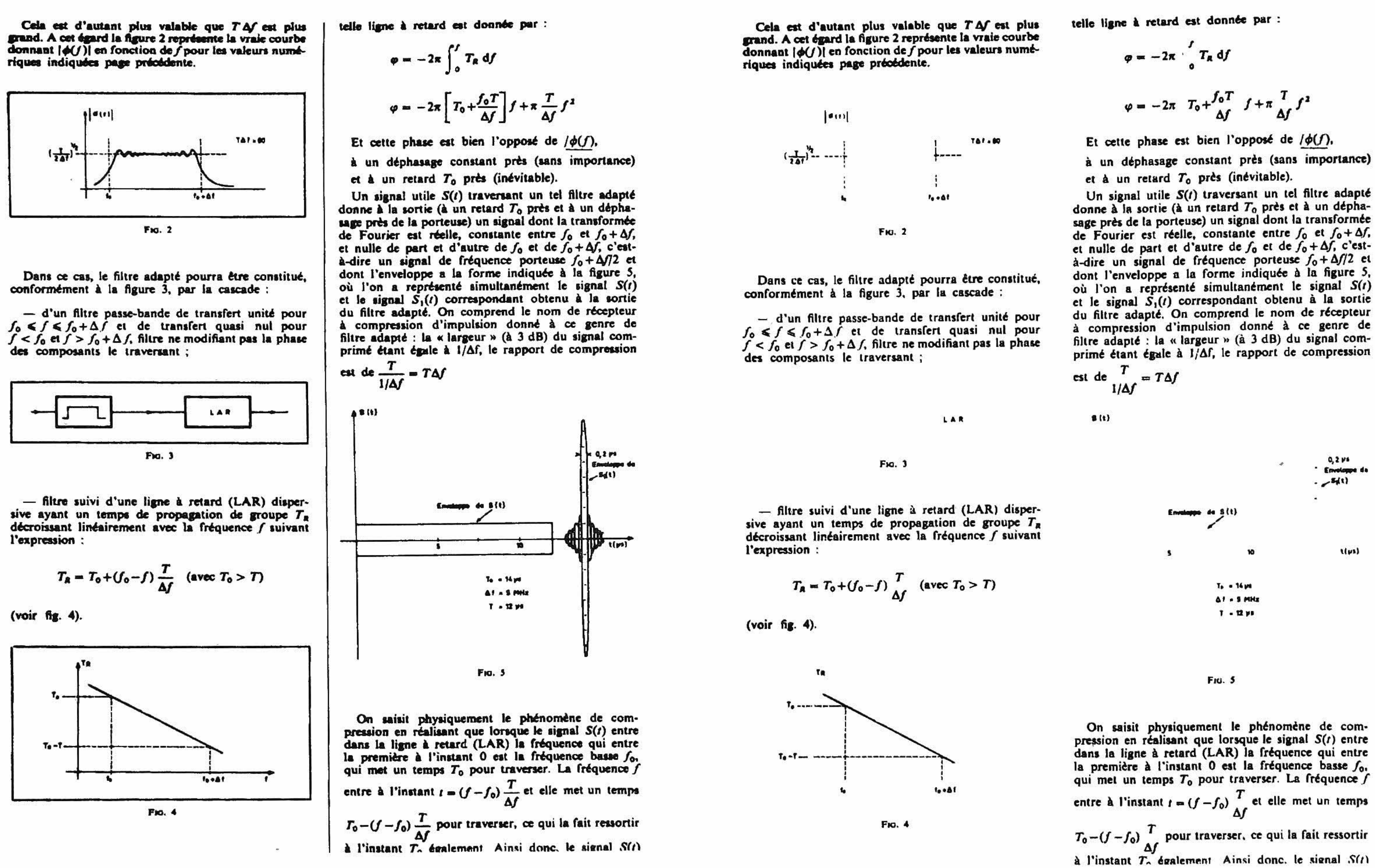


Fig.4-70 Result of the distinction between characters and graphs (CCITT NO.2).



Cela est d'autant plus valable que $T\Delta f$ est plus grand. A cet égard la figure 2 représente la vraie courbe donnant $|\phi(f)|$ en fonction de f pour les valeurs numériques indiquées page précédente.

FIG. 2

Dans ce cas, le filtre adapté pourra être constitué, conformément à la figure 3, par la cascade :

— d'un filtre passe-bande de transfert unité pour $f_0 \leq f \leq f_0+\Delta f$ et de transfert quasi nul pour $f < f_0$ et $f > f_0+\Delta f$, filtre ne modifiant pas la phase des composants le traversant ;

FIG. 3

— filtre suivi d'une ligne à retard (LAR) dispersive ayant un temps de propagation de groupe T_R décroissant linéairement avec la fréquence f suivant l'expression :

$$T_R = T_0+(f_0-f)\frac{T}{\Delta f} \quad (\text{avec } T_0 > T)$$

(voir fig. 4).

FIG. 4

telle ligne à retard est donnée par :

$$\varphi = -2\pi \int_0^f T_R \, df$$

$$\varphi = -2\pi\left[T_0+\frac{f_0 T}{\Delta f}\right] f+\pi\frac{T}{\Delta f}f^2$$

Et cette phase est bien l'opposé de $/\phi(f)$,

à un déphasage constant près (sans importance)

et à un retard T_0 près (inévitable).

Un signal utile $S(t)$ traversant un tel filtre adapté donne à la sortie (à un retard T_0 près et à un déphasage près de la porteuse) un signal dont la transformée de Fourier est réelle, constante entre f_0 et $f_0+\Delta f$, et nulle de part et d'autre de f_0 et de $f_0+\Delta f$, c'est-à-dire un signal de fréquence porteuse $f_0+\Delta f/2$ et dont l'enveloppe a la forme indiquée à la figure 5, où l'on a représenté simultanément le signal $S(t)$ et le signal $S_1(t)$ correspondant obtenu à la sortie du filtre adapté. On comprend le nom de récepteur à compression d'impulsion donné à ce genre de filtre adapté : la « largeur » (à 3 dB) du signal comprimé étant égale à $1/\Delta f$, le rapport de compression

est de $\frac{T}{1/\Delta f} = T\Delta f$

FIG. 5

On saisit physiquement le phénomène de compression en réalisant que lorsque le signal $S(t)$ entre dans la ligne à retard (LAR) la fréquence qui entre la première à l'instant 0 est la fréquence basse f_0, qui met un temps T_0 pour traverser. La fréquence f

entre à l'instant $t = (f-f_0)\frac{T}{\Delta f}$ et elle met un temps

$T_0-(f-f_0)\frac{T}{\Delta f}$ pour traverser, ce qui la fait ressortir

à l'instant T_0 également. Ainsi donc, le signal $S(t)$

Fig.4-71 Result of the distinction between characters and graphs (CCITT NO.5).

IV-5-3 Extraction of the Feature Points

Hereafter we restrict our discussion to the diagrams without characters, i.e. graphs which are the results of the distinction process described in the previous section. So, our graphs consist of two components, symbols and lines to connect them. Symbols are not as complex as the characters, therefore we treat them as a collection of lines. In other words, the symbols and the lines are not distinguished at all, and all of them are treated on the same level.

From this standpoint, the graph is considered to be constructed by lines, and there exist points of intersection of the lines, bending points of the lines and ending points here and there. Here, we define the feature points, which consist of three kinds as follows.

(1) crossing point --- the point of intersection of two or more lines
(2) bending point --- the point where the direction of the line is changed
(3) ending point --- the point where the line begins or terminates

Fig.4-72 illustrates examples of these three kinds of the feature points. Using them, the diagrams are described by the information about the feature points and the connections between them.

The question may arise what to do if there is a graph without any these feature points, like a circle etc. In our process, a loop without a relatively long line is eliminated as a character. The other loops are neither symbols nor lines which connect them, so that they are not included in our objective graphs.

In this section, we describe the algorithms for extracting these feature points. Our process works on a SYM-picture and on the results of the automaton described in section IV-3-3. That is, each symbol of a SYM-picture has values, not only its identifier of legal patterns, but also the two widths of the line elements it belongs to, in the horizontal and vertical directions. These values are re-calculated as diagrams without characters in order to make them more suitable. According to this, the threshold values V_w , V_l and H_w , H_l are a little changed.

Since this width information is estimated only in the horizontal and vertical directions, we must explain the process to extract the feature points in two cases, i.e. one case when the feature points are defined only by nearly horizontal and nearly vertical

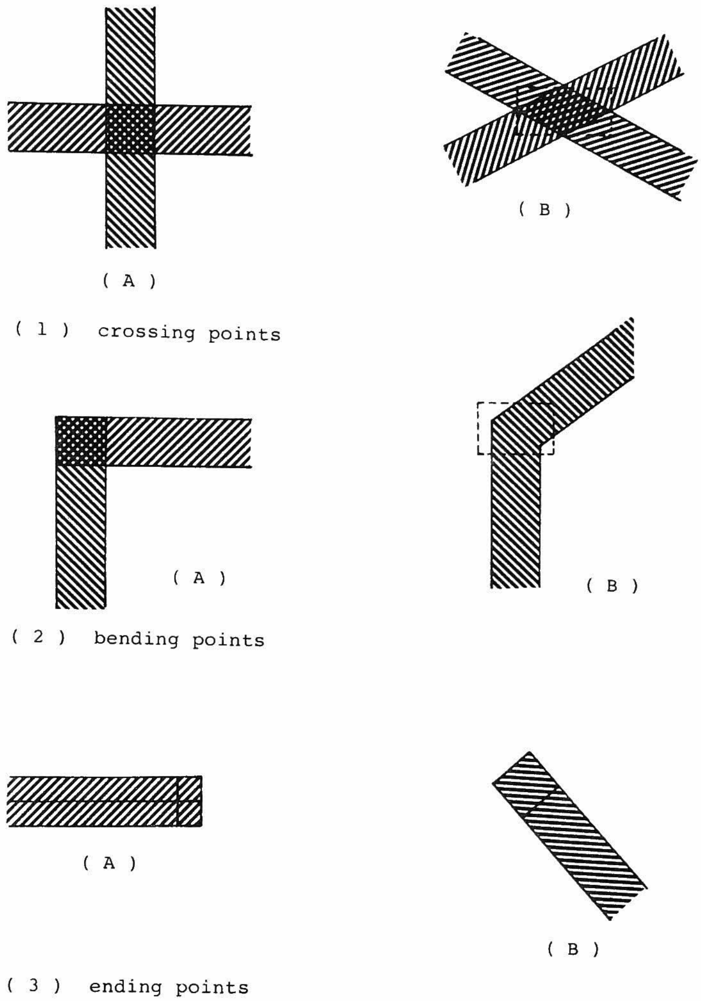


Fig.4-72 Feature points.

lines, and the other case when they include slant lines. In both cases the feature points are extracted as MBRs (Minimum Bounding Rectangular).

(A) For those feature points which are the processing results of only nearly horizontal and nearly vertical lines. Examples corresponding to each case are shown in Fig.4-72(A).

(1) crossing points

The candidate symbols have the property that both widths are greater than V_l

and H_l, respectively. The MBRs to be defined by them are extracted as the cross points.

(2) bending points

In this case, the bending points are not distinguished from the crossing points, so that, the same process is applied to extract them.

(3) ending points

These points are specified by having the W patterns in the direction of the lines. The candidate symbols are identified by the condition that the symbols have at least three W patterns in their 8-neighbours in the direction of the line as shown in Fig.4-72(3)(A), where the direction of the line is the direction with width greater than V_l or H_l.

(B) For other feature points

In this case, the process is not straightforward, because we only have the information observed in the horizontal and vertical directions. The process for handling slant lines is a little complex and heuristic. The procedure is illustrated in Fig.4-73.

(1) crossing points

The points of intersection of slant lines and nearly horizontal or nearly vertical lines are extracted by the same process (i) in the case of the bending points. The crossing points with both slant lines are extracted by applying the process four times (described in the next (ii)) to extract the bending points as is shown in Fig.4-73(1).

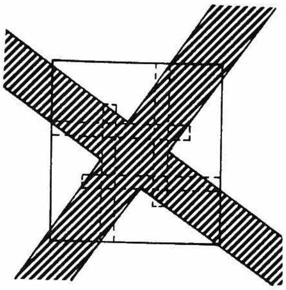
(2) bending points

The process for the bending points of both slant lines is a little different from that of slant lines and nearly horizontal and nearly vertical lines.

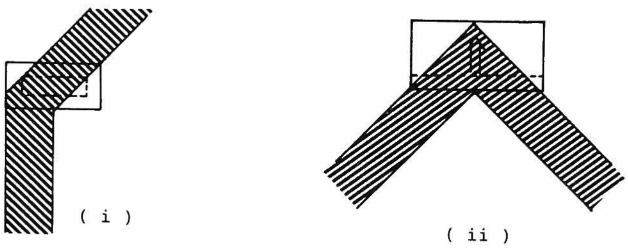
(i) In the case that one of the two lines which makes the bending point is nearly horizontal or vertical.

The candidate symbols of the bending points are identified by checking if the line width is greater than the threshold H_w or V_w. The MBR is derived from these candidate symbols (see the Fig.4-73(2)(ii)).

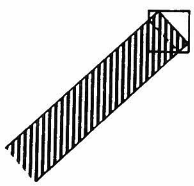
(ii) In the case that both of the lines making bending points are slant ones.



(1) crossing point



(2) bending points



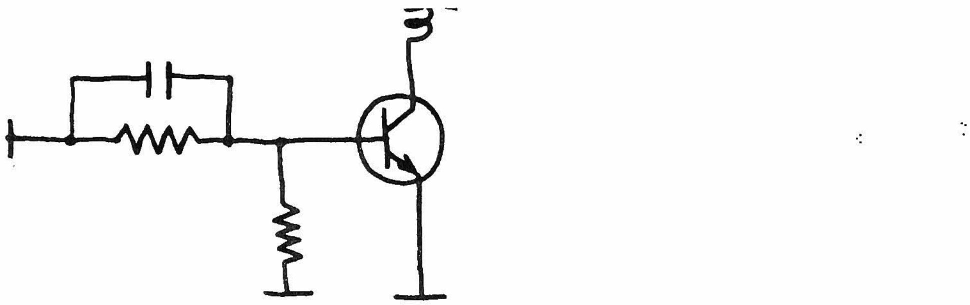
(3) ending point

Fig.4-73 Procedure to extract the feature points (the case (B)).

The candidate symbols have width about two times as large as the threshold H_w or V_w and at the same time, the line elements with that width must have symbols with directions that make an acute angle with each other. The directions of legal patterns themselves are described in section IV-3-2-2. The MBR is defined to extend the candidate symbols to the top of the angle as shown in Fig.4-73(2)(ii).

(3) ending points

The process is almost the same as in the previous case, though the estimation of the direction is a little difficult.



(a) original data (b) extracted feature points

Fig.4-74 Examples of extracted feature points.

Fig.4-74 is an example of results derived after the process described in this section. From this figure, we can see that the feature points of the case (1) are extracted correctly, but those in (2) are not done so well. The key of this process is the setting of the four thresholds, therefore we re-calculate these values for the diagrams without characters. Even doing so, there are some points not extracted well. The main reason is in the heuristics introduced here.

The feature points extracted here are only MBRs. It is difficult to get the position of the feature points accurately, because the lines have width. For our purpose here, we handle hand sketched line drawings, so that accurate positions themselves are not well defined in a strict sense. Therefore, we think it may be sufficient to extract the features points such as the MBRs.

The results are, as a whole, good, but failure to extract these feature points means the failure of the whole process. It is a problem for the future to make this kind of process free from these heuristics.

IV-5-4 Distinction between Curves and Straight Lines

After extracting the feature points, the next step is to get information about the connections between them. The automaton described in section IV-3-3-2 can obtain the connective information at the same time as the widths of the line elements. Using this information, we are able to know which feature points are on the same connected components. Upon those points the tracing process is applied to decide which feature point is connected to another. In consequence of this process, we can obtain topological information about the given diagrams.

It is not sufficient to reconstruct the original diagram only from the topological one, because some information is missing about the kind of lines which connect the feature points. Therefore, we must distinguish whether connection lines are straight or not. Nagura [56] has developed a method to approximate hand sketched line drawings into straight lines and circular arcs. He obtained good results for the data that consist of simple arcs defined in **Definition 4-2** in section IV-2-1.

Our diagrams consist of digital lines with width, so that there must be another method to do almost the same thing. Still more, our process works on a SYM-picture, not on the original one. To take advantage of being represented by a SYM-picture, we only investigate both end symbols of all the line elements belonging to the line. Here, we need finer grouping of the legal patterns as stated in Fig.4-39 of section IV-3-2-2. They are shown in Fig.4-75, and each corresponds to one of 8-directions.

The straight lines in digital space in the form of simple arcs have at most two kinds of chain codes [57,58]. Taking this fact into consideration, we can distinguish between straight lines and curves from statistics about the direction in Fig.4-75 of boundary symbols of the line elements. In other words, the directions of boundary symbols are at most two if they are in straight lines. In the real world, there exists noise, so that we check this property by taking a histogram of the direction of boundary symbols. Fig.4-76 explains this procedure. As shown in this figure, the boundary symbols are decided respectively in each direction, and the boundary symbols are devided into two classes, i.e. the symbols of the right boundary and the left one. We can, therefore, make four histograms for these boundary symbols if we want. Among them, cosidering the

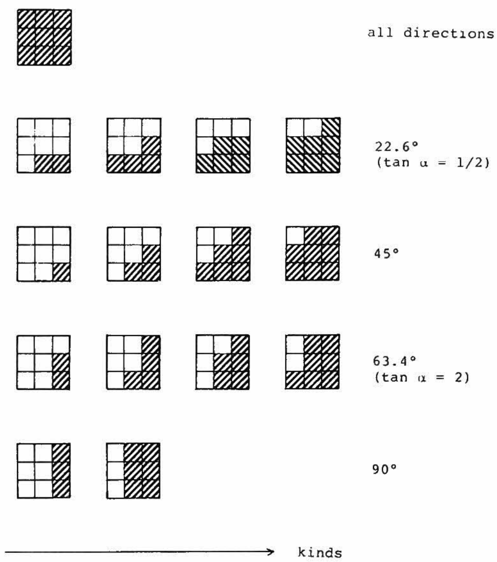


Fig.4-75 Direction of legal patterns.

approximate direction of the line, two of them are selected in order to decide whether the line is straight or not. For example, if the line has nearly vertical direction, the histograms for horizontal boundary symbols are selected for this decision.

In the case that the lines are decided to be straight, the width of the line is calculated. As shown in Fig.4-77, the widths estimated from both nearly horizontal and nearly vertical direction can be used. The direction of that line is estimated from the position of the feature points located at both ends of that line. To decrease errors in the estimation, the average width of the dominant direction is calculated, and from this value the final estimation of that line width is obtained.

On the other hand, curves are treated in a little more complex way. Since our objective diagrams are hand sketched, and contain curves with width, the method to represent curves in the form of mathematical equations is not so desirable. Here, it seems good enough to approximate curves by several straight lines. To do so, we introduce auxiliary feature points, and the procedure becomes almost the same as that for straight lines.

IV-5-5 Reconstruction of the Images

Based on the procedures described in the previous sections, we construct descriptions of the graphs in the original images. A description corresponds to a component, i.e. one connected graph.

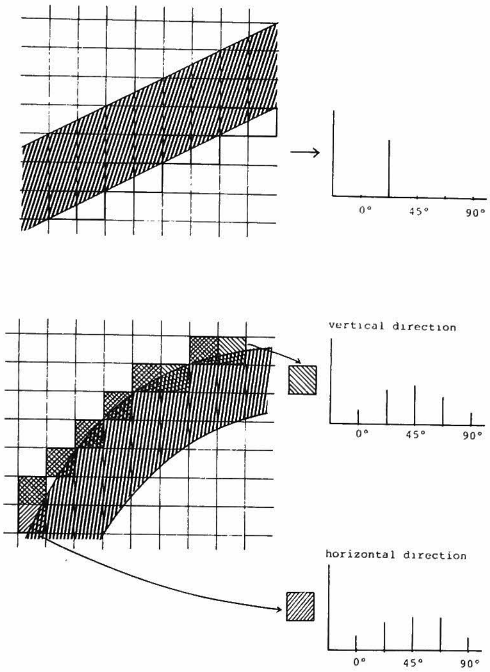


Fig.4-76 Procedure to distinguish between curves and straight lines.

The graph representations are generally used to represent topological information. Here we use them, and the feature points are represented by their nodes, and the connections between them by their arcs. It is possible to have more than one arc between the same nodes, so that the arcs have labels. Actually, multi-linked list representation is used in our system.

A concrete format of the lists for the feature points and auxiliary ones are shown in Fig.4-78(a) and (b), respectively. The direction of the arc is not so important, so that the representation is not a doubly linked one. We have the lower layer link list for the arcs corresponding to

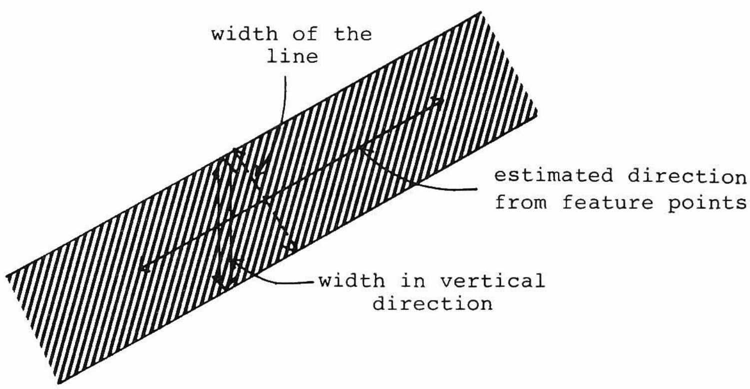


Fig.4-77 Estimation of line width.

curves. The list shown in Fig.4-78(a) may have link parts with the same number of arcs as the corresponding feature point.

As a simple example, the whole representation corresponding to a simple diagram shown in Fig.4-79(a) is shown in Fig.4-79(b). It can be seen that this representation has

Node ID	kind of feature points	coordinate (x, y)	L I N K	kind of arcs	width or sublink

(a) node for feature points

node ID	coordinate (x, y)	L I N K	width

(b) node for auxiliary points

Fig.4-78 Format of multi-linked list for description.

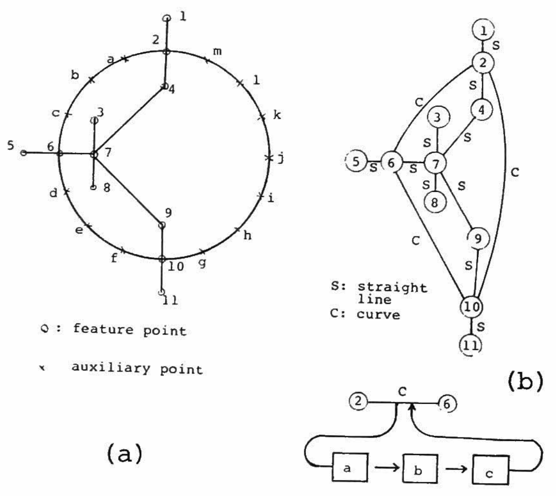
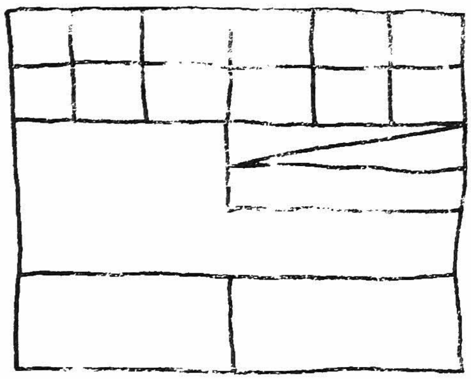


Fig.4-79 Example of description.

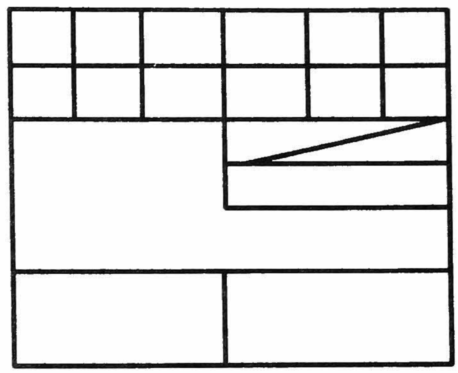
enough information to reconstruct the original image.

The extracted positions of the feature points are not as accurate since the original diagrams themselves are not well represented. And still worse, the lines have width. Therefore, it is not expected to obtain a good reconstructed image without any processing of this representation. We apply a simple process of alignment of straight lines to this representation. That is, setting a threshold value to distinguish nearly horizontal lines into horizontal ones, the positions of the feature points are adjusted.

Original diagrams, and the reconstructed images after applying this simple process, are shown in Fig.4-80(A) and (B). (A) is a table written by hand with a pencil. There is a point where this process on that graph representation does not work well. It is due to the original diagram distortion from the hand sketching. The data (B) is a part of CCITT's test

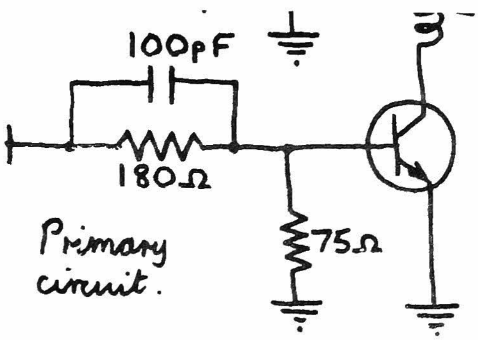


(a) original image

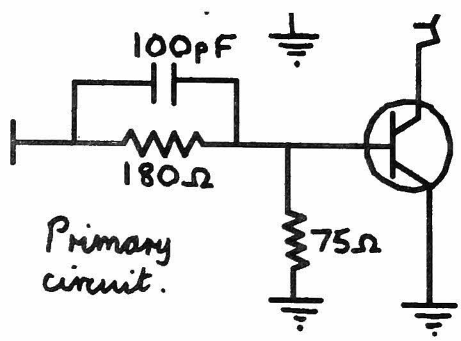


(b) reconstructed image

(A)



(a) orignal image



(b) reconstructed image

(B)

Fig.4-80 Examples of reconstructed images.

chart NO.2. There is a point not reconstructed well; the arrow in the transister symbol. It is due to the process i.e. the failure of extracting the feature point of the ending of the arrow, which seems to be difficult from human observation of the original. This will be solved by processing a little more on that representation.

IV-5-6 Discussion

In this section IV-5, we describe the procedure for reconstructing hand sketched diagrams on the basis of MOLD Theory. The whole procedure is summarized in the following steps.

(1) The hand sketched diagrams are transcripted into a SYM-picture according to their qualities.

(2) The automaton is applied on the SYM-picture in order to get the width information about digital lines included and the connection information about them.

(3) The process to eliminate the characters included is applied on the basis of the information obtained at the previous step (2).

(4) The feature points, i.e. the crossing points, the bending points, and the ending points are extracted from the resultant diagram after the process of eliminating the characters.

(5) Considering the information of connections among lines, a list representation of the diagram is obtained.

(6) After applying some simple process to this representation, the reconstructed image is composed.

The results as a whole are good as considering the ambiguity of the hand sketched diagrams themselves. The simplicity of our process leads us to conclude that the representation based on legal patterns is very suitable for this kind of process. Doing the same process on the original image might cost much more time and need much more complex algorithms.

Indeed, there remain some problems to be solved, the most intrinsic one being how to handle the parts which are not lines but regions. For example, there is a symbol for diode in electric circuits. These parts may be found from the width information on the SYM-picture.

The followings are problems of the process described in this section.

(1) The elimination procedure for the characters is based on the connectivity among the lines. In this case, the problem of handling the dot lines arises.

(2) The breaking of lines due to noise is also a problem, but MOLD Theory has the restriction of the sampling process, so that this case will not occur as frequently. The problem is intrinsically the same as (1).

(3) The errors of the estimation are also a problem though the original diagram is hand sketched. The process on the representation must check a larger area in order to align the lines. A big problem of this phenomenon is that the lines are separated by the feature points in this description.

(4) The accuracy of describing the curves must be considered in more detail, because at a glance the reconstructed curves are a little strange.

This research is only aimed at constructing the description of diagrams without the characters. Therefore, the work to process this description are almost left unsolved. There is a possibility to transform this description to that used in the field of computer graphics. In order to do so, a much more complex process with the ability of symbol recognition is needed. The tiresome work of inputing graphical data to computers will be removed after the success of these research. Curves will be taken as a very important part of symbol recognition for the kind of diagrams treated here.

IV-6 Concluding Remarks

In this chapter, the legal patterns which represent pattern generating constraints in a concrete form are proved to be very useful in three applications, not only in information processing, but in digital communication. MOLD Theory is tested by various experiments, which show the model proposed by this theory is very suitable for the real world.

Legal patterns are especially useful for deciding the image quality because they represent the "correct" lines, and the set of legal patterns is very small, which causes information compression of the images. It is true that it is an approximation to represent bi-level line drawings as only the legal patterns, but the quality of the resultant image is not worse than the original because of the noise removal ability of the legal patterns themselves. The small amount of information makes the coding process on a LSC-picture much better from the point of the compression ratio. It is worth while commenting that the relations between adjacent symbols are available for error detection and correction, which will become a new aspect of digital communication. A SYM-picture is also useful for the processing of hand-sketched diagrams. This problem will become more and more important in practical use.

The underlying concept is that we must take advantage of the pattern generating constraints when processing the patterns of various kinds of line drawings. The real world patterns are generated by people in a society, and they are subject to the law of human society as well as the law of nature. Keeping this fact in mind, there must exist a field in which the pattern generating constraints are available in the actual process. Our world of line drawings is one of the tasks of this kind.

Note the way MOLD Theory is constructed. First, it assumes a hypothesis. Next, the theory is constructed on the basis of that assumption, and this theory is tested in the real world. These steps are exactly the same as taken in science, e.g. in physics, mathematics etc. The field of information processing is, we believe, also science.

There are some problems as to MOLD Theory, though.

(1) As the line width becomes wider, the efficiency of the legal patterns becomes lower.

(2) At the points where two digital lines cross in acute angle, the sharpness is relatively lost in our representation.

(3) In the case where lines with one pixel width exist, representation is not possible. In practice, it seldom occurs.

These are concerned with the resolution of the images. The total system of an image input device including these problems, must be considered in the future.

CHAPTER V

CONCLUSION

V-1 Summary of This Thesis

This thesis has been devoted to the description of the basic relations between real world pictorial patterns and their generating constraints. Since the patterns are a concrete media of communication, we do not have to make any distinction between the field of information processing and digital communication. Like the premise of communication, constraints used in the pattern generating process must be used in the process of disposing of patterns. This idea is applied more easily in the world of man-made things than that of natural things.

This thesis contains two applications i.e. a depth measurement method and MOLD Theory, and they are summarized briefly.

(1) A method of time-coded parallel planes of light for depth measurement

We transmitted many planes of light to the objective space simultaneously. To identify each plane of light, we assigned a unique code word with equal length to it, and the code word controlled the "on" and "off" of the plane of light in the time domain. Since the scene with many slit images was considered to be a real world pattern, the pattern generating constraint was regarded as this code. The properties of codes took an important role for the process. Giving the code some redundancies, the process became stronger for noise though it cost more computation time.

This is a favorable property of our system and a very important feature. That is, if the circumstance is good enough, there is no need for the real world patterns to have a rather complex "structure". However, in other cases, we must put a strong structure to the real world patterns in order to distinguish them from noise. If we proceed in this direction, the next problem will be to decide whether the circumstance of the system is favorable or not in the time needed to measure the depth. Accommodating this facility, our method of depth measurement will become more practical. Research of this kind is left for the future.

(2) In the case of bi-level line drawings

(i) MOLD theory

The world of this theory was the figures of lines. The real world patterns were considered to be any line drawings in an analog world. The image obtaining constraint was the restriction of the sampling process (analog to digital conversion), that the sampling intervals must be less than half those specified by the sampling theorem.

The assumption of this theory was that the size of the unit mesh, which was the observation unit, was less than or equal to the width of the digital lines. Based on this assumption, the legal patterns were derived theoretically as a kind of representation of the pattern generating constraints. Using these legal patterns only, we could describe the bi-level line drawings satisfying the sampling condition and the assumption. Strictly speaking, the process was a kind of approximation, though the quality was not worse than that of the original. This fact was tested experimentally and the new representations such as a SYM-picture and an LSC-picture were proposed.

(ii) Adaptive system for noise elimination

Since legal patterns were defined on the basis of the pattern generating constraints, they represented the "correct" lines in a sense. We decided the image quality according to the statistics of legal patterns in order to obtain rough information on the following process. The properties of the statistics of legal patterns were proved to have many possibilities, and research concerned with them was left for the future.

(iii) Redundancy reduction coding

LSC-pictures had less information than the original. This was the reason why we tried to do coding on LSC-pictures. The compression ratio of our coding methods were almost as high as that of Modified READ, and as to other properties they are almost the same. The advantage of our coding methods is that ours are very suitable for information processing, and the relations between symbols were available not only for compressing the amount of information, but for the possibility of error detection and correction. Theoretical studies on how to take advantage of the relations in the theory of coding were left for the future.

(iv) Reconstruction of hand sketched line drawings

The fitness of SYM-pictures for information processing was experimentally investigated. Characters were eliminated from the objective line drawings by means of the parameters of connectivity and simpleness. For the residues, the description based on the feature points and connective information between them was constructed. After a simple disposal on this description, the reconstructed image was fed into an output device. These processes are rather simple, and faster due to the process being on SYM-pictures. In future, we will develop a more complex process with the power of recognition on that description to transform it to the form of the data used in computer graphics.

V-2 Pattern Generating Constraints

Real world patterns are concrete representations of information. From this aspect, they are generated by people using a tool under the law of nature and of human culture. For the process of disposing of the patterns of this kind, the constraints to generate them may take an important role. In this thesis, we mainly discussed these constraints, i.e. pattern generating constraints.

Fig.5-1, again, shows the communication concept by means of real world patterns. It contains three kinds of constraints, pattern generating constraints, image obtaining

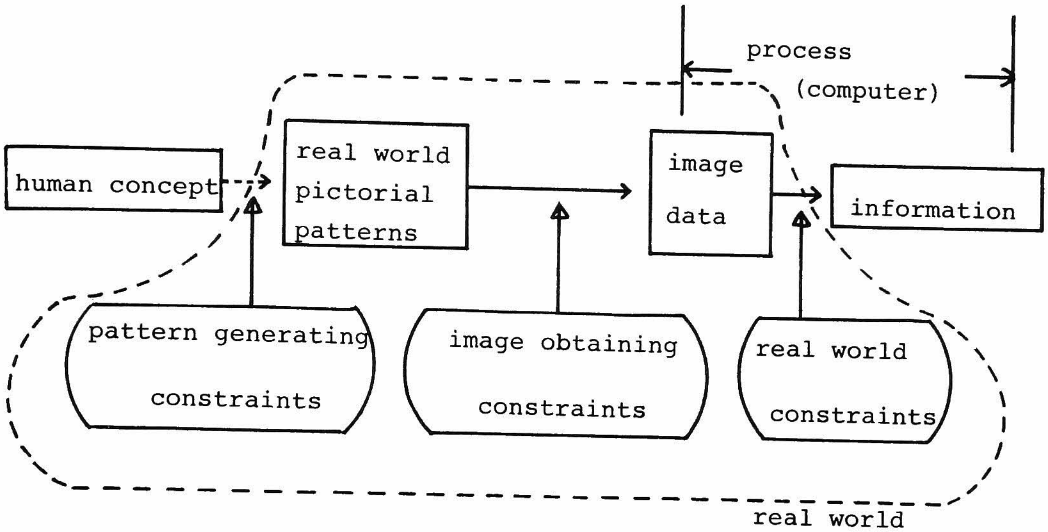


Fig.5-1 Pattern generating constraints.

constraints and real world constraints. If we make the rough correspondence between these constraints and the level of the image processing, pattern generating constraints belong to the signal level (see Fig.5-1). The constraints in the signal level are especially useful for such work as noise elimination and rough pre-classification for more detailed processing. Even if the pattern generating constraints are introduced in a process, it will not cause any loss of generality.

In future, image processing systems must accept various kinds of input data. In this case, the importance of pattern generating constraints will become greater. Further studies as to this kind of constraint will bring us to the concept of "information filter", which works like a conventional filter. With this filter, the image processing system can select the input data to be disposed and apply the most suitable process. Through some experimental results presented in this thesis, we believe that there must exist pattern generating constraints which are very important and useful, not only in the field of information processing, but also of digital communication.

V-3 Area for Future Works

Some work is left for the future. Specific areas of research are already described in previous sections. Only the key idea, i.e. pattern generating constraints, is commented upon further.

In this thesis, we presented only two worlds able to use this kind of constraint. To prove the existence of the pattern generating constraints, some other task worlds must be established, and the theory and some applications must be studied. This research will lead us to general agreement as to the existence of pattern generating constraints in general.

* General

[1] D. A. Huffman, "Impossible objects as nonsense sentences", Machine Intelligence, vol.6, Edingburgh University Press, pp.295-323, 1971.

[2] M. B. Clowes, "On seeing things", Artificial Intelligence, vol.2, no.1, pp.101-144, 1971.

[3] D. Waltz, "Generating semantic descriptions from drawings of scenes with shadows", The Psychology of Computer Vision, McGraw Hill, pp.19-91, 1975.

[4] B. K. P. Horn, "Understanding image intensity", Artificial Intelligence, vol.8, no.2, pp.201-231, 1977.

[5] M. I. Bernstein, "Computer recognition of one-line, hand-written characters", RAND Corporation Memorandum, RM-3753-ARPA, 1964.

[6] M. Suwa, Y. Shirai, K. Koshikawa, M. Oshima, K. Yamagami and S. Tsuji, "A hardware system of an industrial eye", Journal of the Electrotechnical Laboratory, vol.35, no.3, pp.252-266, 1971.

[7] P. M. Will and K. S. Pennington, "Grid coding: A novel technique for image processing", Proc. IEEE, vol.60, no.6, pp.669-680, 1972.

[8] J. Yamato, "Moire contour map processing", Inf. Process. Japan, vol.20, no.12, pp.1082-1088, 1979. (in Japanese)

[9] S. Yonezawa and Y. Tamamura, "Coded grating method for measuring three dimensional objects", Trans. IECE Japan, vol.J61-D, no.6, pp.411-418, 1972. (in Japanese)

[10] K. Sugihara, "Studies on mathematical structures of line drawings of polyhedra and their applications to scene analysis", Res. Electrotechnical Lab., 800, pp.24-27, 1979. (in Japanese)

[11] "Information processing handbook", pp.545-547, Information Processing Society of Japan, 1980. (in Japanese)

[12] M. D. Altschuler, B. R. Altschuler and J. Taboada, "A laser-optic system for rapid 3-D topographic mapping of surfaces", Medical Image Processing Group Tech. Report, no.MIPG51, State Univ. of New York at Buffalo, 1981.

[13] T. Ueda and M. Matsuki, "Time sequencial coding for three-dimensional measurement and its inplementation", Trans. IECE Japan, vol.J64-D, no.8, pp.780-787, 1981. (in Japanese)

* Bi-level images

[14] M. Murao and T. Sakai, "Extraction of the structure for documents", Proc. of the 21th Joint Convention of the IPSJ, pp.857-858, 1980. (in Japanese)

[15] H. Makino and Y. Akada, "On bi-level rendition for a document image including half-tone pictures", Trans. IECE Japan, vol.J65-D, no.3, pp.307-314, 1982. (in Japanese)

[16] A. Rosenfeld and A.C.Kak, "Digital picture processing", Academic Press, pp.335- , 1976.

[17] A. Rosenfeld, "Arcs and curves in digital picture", Journal of ACM, vol.20, no.1, pp.81-87, 1973.

[18] R. C. Gonzalez and P.Wintz, "Digital image processing", Addison-Wesley Publishing Company, 1977.

[19] J. F. Jarvis, "A method for automating the visual inspection of printed wiring boards", IEEE Trans. of Pattern Analysis and Machine Intelligence, vol.PAMI-2, no.1, pp.77-82, 1980.

[20] S. Yokoi, J. Toriwaki and T. Fukumura, "Topological properties in digitized binary picture", Trans. IECE Japan, vol.56-D, no.11, pp.662-669, 1973. (in Japanese)

[21] J. F. O'Callaghan and J. Loveday, "Quantative measurement of soil cracking patterns", Pattern Recognition vol.5, no.2, pp.255-264, 1971.

[22] D. Ting and B. Prasada, "Digital processing techniques for encoding graphics", Proc. IEEE, vol.68, no.7, pp.757-769, 1980.

* Image Quality and Noise Removal

[23] A. Lev, S. W. Zucker and A. Rosenfeld, "Iterative enhancement of noisy images", IEEE Trans. on Systems, Man and Cybernetics, vol.SMC-7, no.6 pp.435-442, 1977.

[24] W. K. Pratt, "Generalized Wiener filtering computation techniques", IEEE Trans. Computers, vol.C-21, pp.636-692, 1972.

[25] N.E.Nahi and T.Assefi, "Bayesian recursive image estimation", ibid., pp.730-734, 1972.

[26] A. Habibi, "Two-dimensional Bayesian estimate of images", Proc. IEEE, vol.50, no. , pp.878-883, 1972.

[27] G. P. Dinneen, "Programming pattern recognition", Proc. Western Joint Computer Conference, pp.94-100, 1955.

[28] T. Shimoe, T. Saito and Y. Hoshino, "Two-dimensional smoothing of noise in image by Kalman filtering", Trans. IECE Japan, vol.J61-D, no.8, pp.541-548, 1978. (in Japanese)

[29] T. Hoshino and S. Kawakubo, "Notch elimination method for binary figures using run-length coding", Trans. IECE Japan, vol.J65-D, no.1, pp.32-39, 1982. (in Japanese)

[30] T. S. Huang and H. U. Koller, "Coding of multilevel graphics", IEEE Trans. on Communication, vol.COM-23, no.6, pp.598-606, 1975.

[31] R. B. Arps, R. L. Erdmann, A. S. Neal and C. E. Schlaepher, "Character ligibility versus resolution in image processing of printed matter", IEEE Trans. on Man-Machine Systems, vol.MMS-10, no.3, pp.66-71, 1969.

* Digital Communication

[32] M. Z. Skolnik, "Radar handbook", McGraw-Hill Book Company, 1970.

[33] R. A. Scholtz, "The spread spectrum concept," IEEE Trans. on Communication, vol.COM-25, no.8, 1977.

[34] W.F.Utlaut, "Spread-spectrum principles and paossible application to spectrum utilization and allocation", Telecommunication journal, vol.45, no.1, pp. , 1978.

[35] H. Tominaga, T. Nakagawa and T. Sanada, "An associative coding on two-dimensional block pattern in facsimile", Technical Report of the Professional group on Image engineering of IECE Japan, IE74-79, 1974. (in Japanese)

[36] D. R. Knudson, "Digital encoding of newspaper graphics", Report ESL-R-616, Electronic System Laboratory, MIT, 1975.

[37] T. S. Huang, "Coding of two-tone images", IEEE Trans. on Communications, vol.COM-25, no.11, pp1406-1424, 1977.

[38] Y. Yasuda, "Overview of digital facsimile coding techniques in Japan", Proc. IEEE, vol.68, no.7, pp.830-845, 1977.

[39] R. B. Arps, "Bibiliography on digital graphic image compression and quality", IEEE Trans. Information Theory, vol.IT-20, no.1, pp.120-122, 1974.

[40] R. B. Arps, "Bibiliography on binary image compression", Proc. IEEE, vol.68, no.7, pp.922-924, 1980.

[41] A. N. Netravali and F. W. Mounts, "Ordering Techniques for facsimile coding: A review", ibid, pp.796-807, 1980.

[42] A. J. Frank, J. D. Daniels and D. R. Unangst, "Progreessive image transmission using a growth geometry coding", ibid., pp.897-909, 1980.

[43] T. Usubuchi, T. Omachi and K. Iinuma, "Adaptive predictive coding for newspaper facsimile", ibid, pp.807-813, 1980.

[44] M. Nakajima and T. Agui, "A delta coding method of contour lines", Trans. IECE Japan, vol.J64-D, no.2, pp.109-115, 1981. (in Japanese)

[45] W. F. Schreiber, T. S. Huang and O. T. Tretiak, "Contour coding of images", Picture Bandwidth Compression edited by T.S.Huang and O.J.Tretiak, Gordon and Breach, 1972.

[46] T. S. Huang and A. B. S. Hussian, "Facsimile coding by skippling white", IEEE Trans. on Communication, vol.COM-23, no.12, pp.1452-1466, 1975.

[47] M. Kunt and O. Johnsen, "Block coding of graphics: A tutorial review", Proc. IEEE, vol.68, no.7, pp.770-786, 1980.

[48] T. Endo and E. Kawaguchi, "Some properties of DF-picture expression and its application to data compression", Trans. IECE Japan, vol.J62-D, no.2, pp.141-148, 1979. (in Japanese)

[49] R. Hunter and A. H. Robinson, "International digital facsimile coding standards", Proc. IEEE, vol.68, no.7, pp.854-867, 1980.

[50] D. Bodson and R. Schaphorst, "Compression and error sensitivity of two dimensional facsimile coding techniques", ibid., pp.846-853, 1980.

[51] W. K. Pratt, P. J. Capitant, E. R. Hamilton and R.H.Wallis, "Combined symbol matching facsimile data compression system", ibid., pp.786-796, 1980.

* Diagrams

[52] K. Ramachandran, "Coding method for vector representation of engineering drawings", ibid., pp.813-817, 1980.

[53] S. Nagata, A. Inoue, T. masui, T. Matsuura and M. Yoshida, "Recognition algorithm for hand printed line drawings on section papers", Technical Report of the Professional Group on Pattern Recognition and Learning of IECE Japan, PRL80-32, 1980. (in Japanese)

[54] H. Terajima, T. Shudo, R. Kawai and J. Watanabe, "A method of removing characters from diagrams", Proc. of the 23th Joint Convention of the IPJS, pp.765-766, 1981. (in Japanese)

[55] O. Iwaki, K. Kubota and S. Ishii, "A characters and graphics extraction method using neighbourhood line density", Technical Report of the Professional Group on Pattern Recognition and Learning, PRL81-81, 1981. (in Japanese)

[56] M. Nagura, "Approximation of line drawings by straight lines and circular arcs", Trans. IECE Japan, vol.J64-D, no.9, pp.839-845, 1981. (in Japanese)

[57] H. Freeman, "On the encoding of arbitrary geometric configuration", IRE Trans. EC-10, pp.260-268, 1961.

[58] H. Freeman, "Boundary encoding and processing", Picture Processing and Psychopictorics (edited by B.S.Lipkin and A.Rosenfeld), Academic Press, pp.241-266, 1970.

[59] S. Shimizu, T. Masui, S. Nagata, T. Matsuura, M. Yoshida and I. Oda, "Automatic input and processing system for printed circuit board design charts", Technical Report of the Professional Group on Pattern Recognition and Learning of IECE Japan, PRL80-53, 1980. (in Japanese)

[60] M. Nagura and Y. Suenaga, "A facsimile graphics editing system having character recognition function", Trans. IECE Japan, vol.J65-D, no.3, pp.402-409, 1982. (in Japanese)

[61] J. F. Jarvis, "The line drawing editor: shematic diagram editing using pattern recognition techniques", Computer Graphics and Image Processing, vol.6, no.5, pp.452-484, 1977.

[62] W. C. Lin and J. H. Pun, "Machine recognition and plotting of hand-sketched line figures", IEEE Trans. on System, Man and Cybernetics, vol.SMC-8, no.1, pp.52-57, 1978.

List of Publications and Technical Reports by the Author

Pubilcations

1. M. Minou, T. Kanade and T. Sakai, "A method of time-coded parallel planes of light for depth measurement", Trans. of IECE Japan, vol.E-64, no.8, pp.521-528, Aug. 1981.

2. M. Minou and T. Sakai, "MOLD(Mesh Oriented Line Drawings') theory --- A representation of the line drawings ---", Trans. of IECE Japan, vol.J65-D, no.7, pp.928-935, July 1982. (in Japanese)

3. M. Minou and T. Sakai, "Classification of the image quality and noise removal method for bi-level line drawings", Trans. Information Processing in Japan, (to be appeared) (in Japanese)

Oral Presentation

4. M. Minou, T. Kanade and T. Sakai, " A method of time-coded parallel lights of slit for the distance measurement", Proc. of the 21th Joint Convention of the IPSJ Japan, pp.875-876, May 1980. (in Japanese)

5. M. Minou, T. Kanade and T. Sakai, "A method of time-coded parallel planes of light for depth measurement", Technical Report of the Professional Group on Computer Vision of IPSJ Japan, CV12-3, May 1981. (in Japanese)

6. M. Minou and T. Sakai, "MOLD Theory ---A representation of the bi-level line drawings ---", Proc. of the 23th Joint Convention of IPSJ Japan, pp.719-720, Oct. 1981. (in Japanese)

7. M. Minou and T. Sakai, "MOLD Theory ---A representation of the line drawings---", Technical Report of the Professional Group on Image Engineering of IECE Japan, IE81-81, Nov. 1981. (in Japanese)

8. M. Minou and T. Sakai, "Noise removal method of bi-level line drawings", Proc. of the 24th Joint Convention of the IPJS Japan, pp.703-704, March 1982. (in Japanese)

9. M. Minou and T. Sakai, "Classification of the bi-level images with their qualities", Technical Report of the Professional Group on Computer Vision of IPSJ Japan, CV17-2, March 1982. (in Japanese)

10. M. Minou and T. Sakai, "Redundancy reduction coding for LSC-pictures", Proc. of the 25th Joint Convention of the IPSJ, pp.873-874, Oct. 1982. (in Japanese)

APPENDICES

Appendix-A Codes, input images and other supplementary results of identification.

Appendix-B Integer sequences corresponding to legal patterns
and supplementary results of SYM-pictures and LSC-pictures

Appendix-C Complete code words

Appendix-A Codes, input images and other supplementary results of identification.

* Codes to be used in the experiments

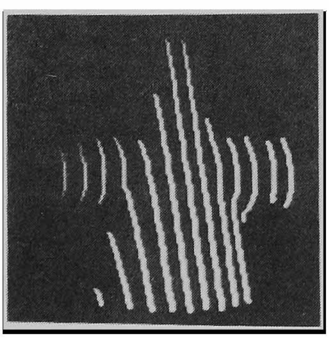
1	1	1	1	1	0
2	0	0	0	1	1
3	1	1	1	0	0
4	0	0	1	0	1
5	1	1	0	1	0
6	0	0	1	1	0
7	1	1	0	0	1
8	0	0	1	1	1
9	1	1	1	0	1
10	1	1	0	0	0
11	0	1	0	0	1
12	1	0	1	1	0
13	0	1	0	1	0
14	1	0	1	0	1
15	0	1	0	1	1
16	1	0	1	0	0
17	0	1	1	0	0
18	1	1	0	1	1
19	1	0	0	1	1
20	0	1	1	0	1
21	1	0	0	1	0
22	0	1	1	1	0
23	1	0	0	0	1
24	0	1	1	1	1
25	1	0	1	1	1

binary code

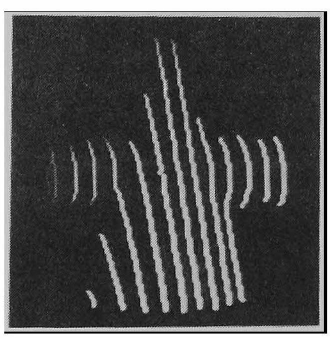
1	1	1	1	1	0	0	0	1	1
2	0	0	0	1	1	1	1	1	1
3	1	1	1	0	0	1	0	0	1
4	0	0	1	0	1	0	1	1	0
5	1	1	0	1	0	0	0	0	0
6	0	0	1	1	0	1	0	0	1
7	1	1	0	0	1	1	1	1	1
8	0	0	1	1	1	1	1	0	0
9	1	1	1	0	1	1	1	0	0
10	1	1	0	0	0	1	0	1	0
11	0	1	0	0	1	0	0	1	1
12	1	0	1	1	0	0	1	0	1
13	0	1	0	1	0	1	1	0	0
14	1	0	1	0	1	1	0	1	0
15	0	1	0	1	1	1	0	0	1
16	1	0	1	0	0	1	1	1	1
17	0	1	1	0	0	0	1	0	1
18	1	1	0	1	1	0	1	0	1
19	1	0	0	1	1	0	0	1	1
20	0	1	1	0	1	0	0	0	0
21	1	0	0	1	0	0	1	1	0
22	0	1	1	1	0	1	1	1	1
23	1	0	0	0	1	1	0	0	1
24	0	1	1	1	1	1	0	1	0
25	1	0	1	1	1	0	0	0	0

single-error-correcting
Hamming code

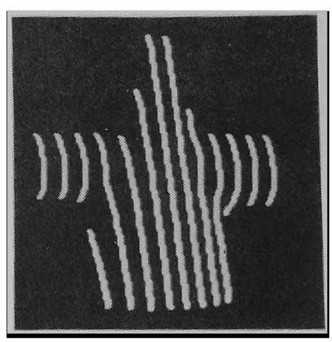
* Results of identification for the imput data shown in Fig.3-8



Algorithm A

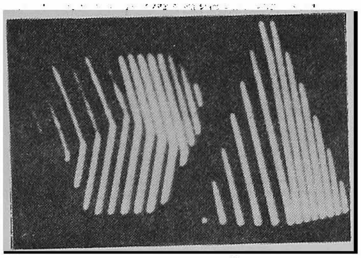


Algorithm B

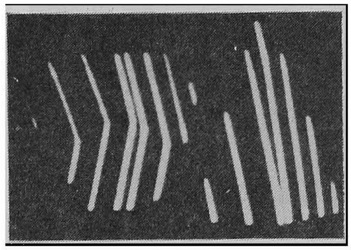


Algorithm C

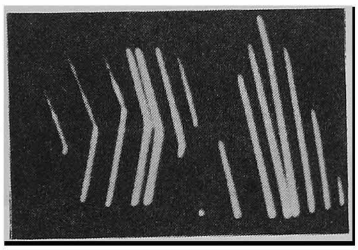
* Input images and the results of identification in dark circumstances



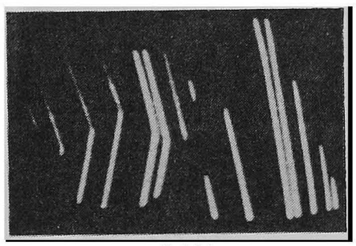
all "on"



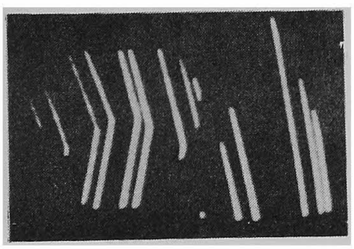
1st bit



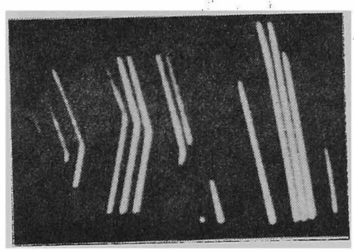
2nd bit



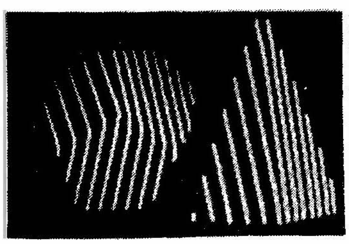
3rd bit



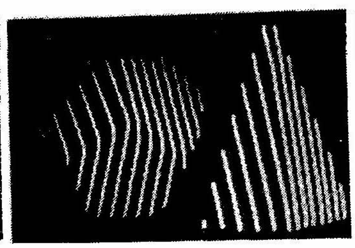
4th bit



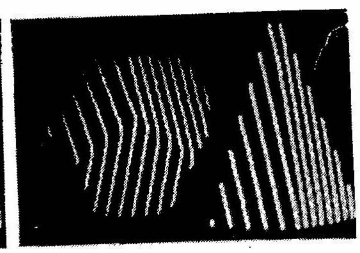
5th bit



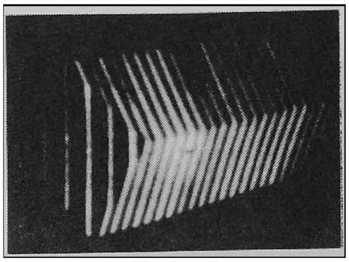
Algorithm A



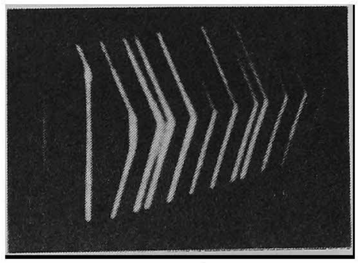
Algorithm B



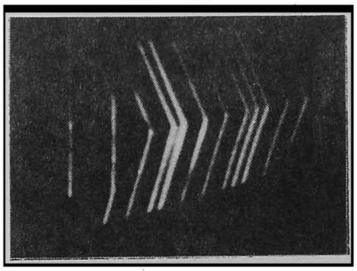
Algorithm C



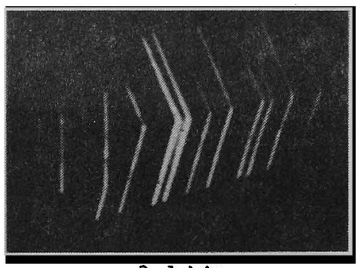
all "on"



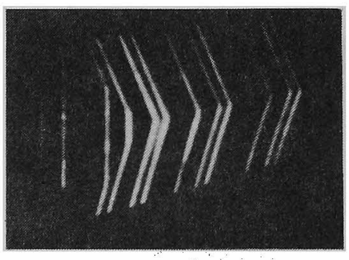
1st bit



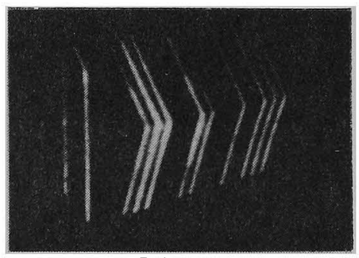
2nd bit



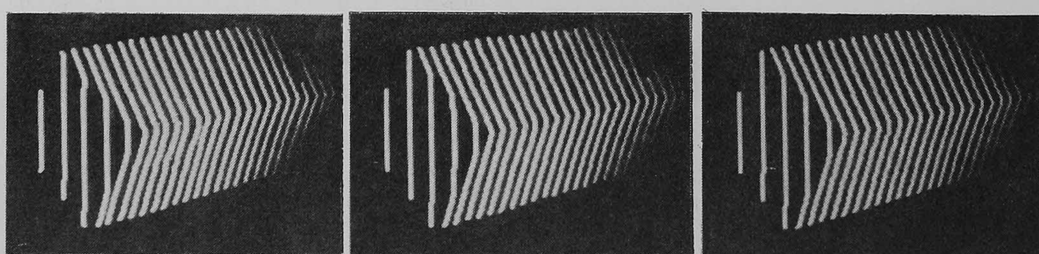
3rd bit



4th bit

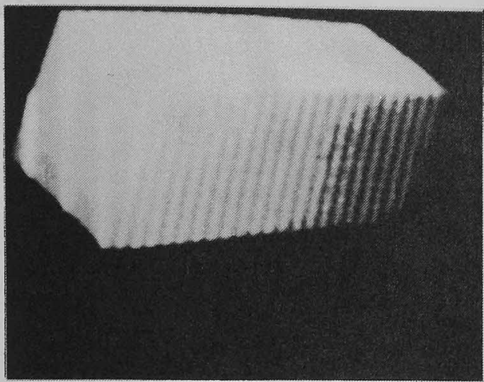


5th bit

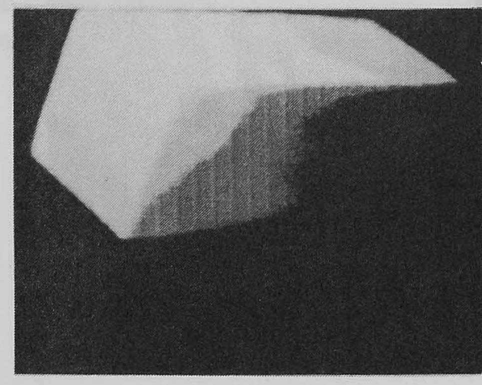


Algorithm A Algorithm B Algorithm C

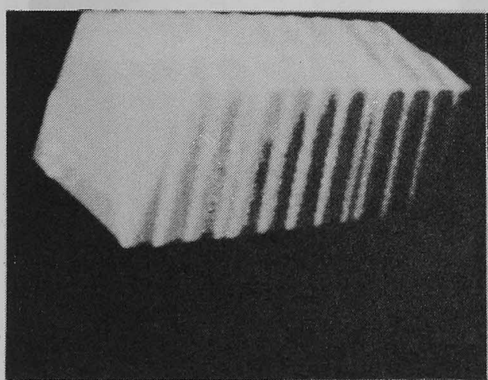
* Input images for the results shown in Figs.3-9 and 3-12



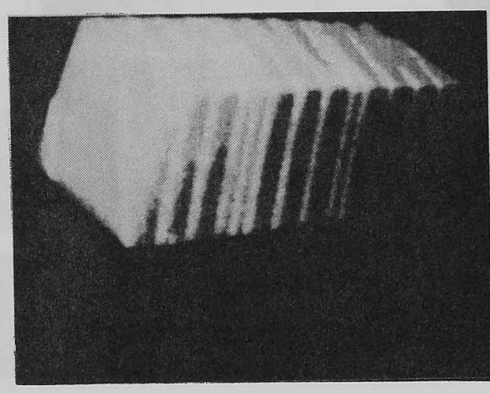
all "on"



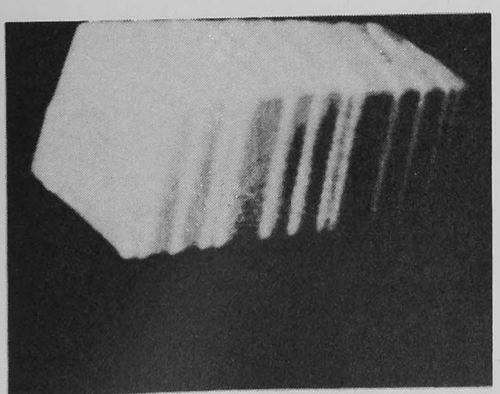
all "off"



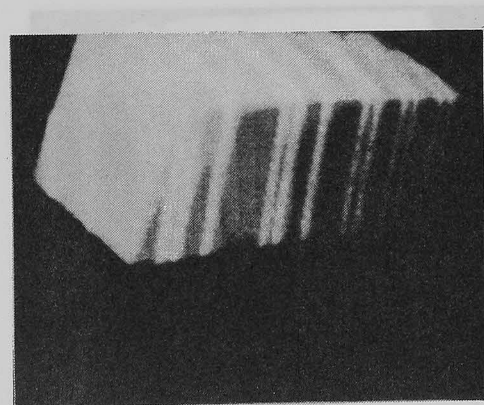
1st bit



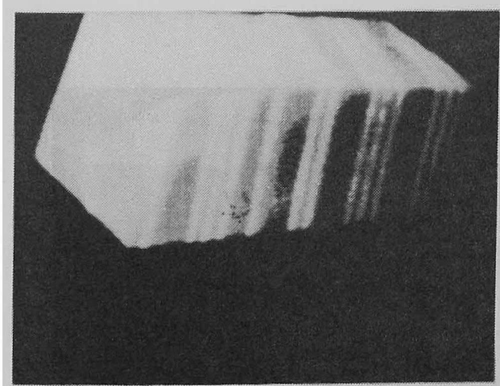
2nd bit



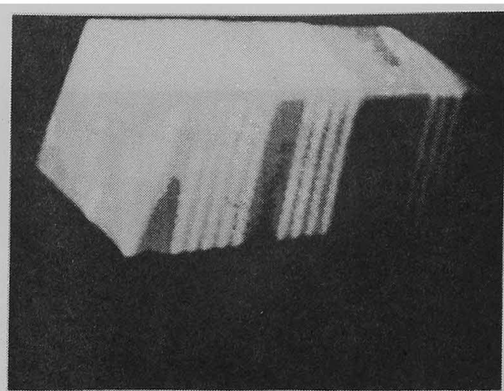
3rd bit



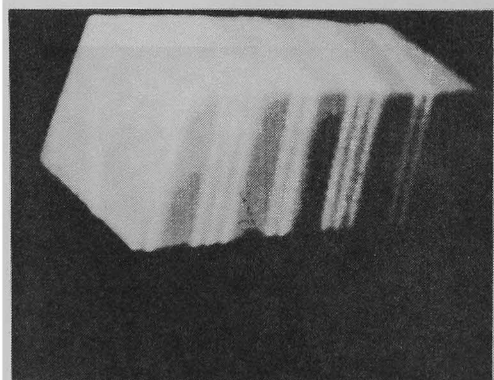
4th bit



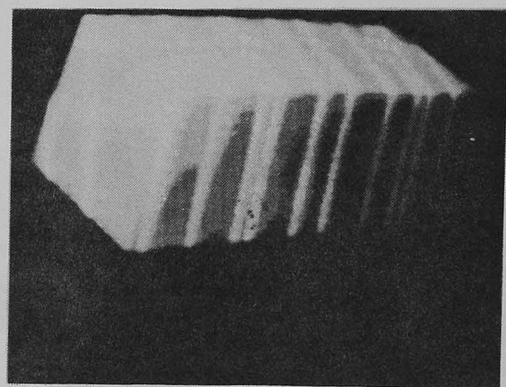
5th bit



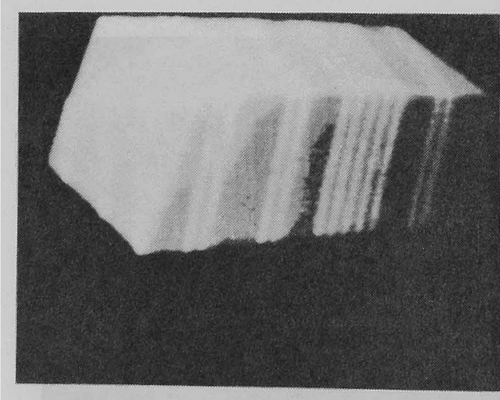
6th bit



7th bit



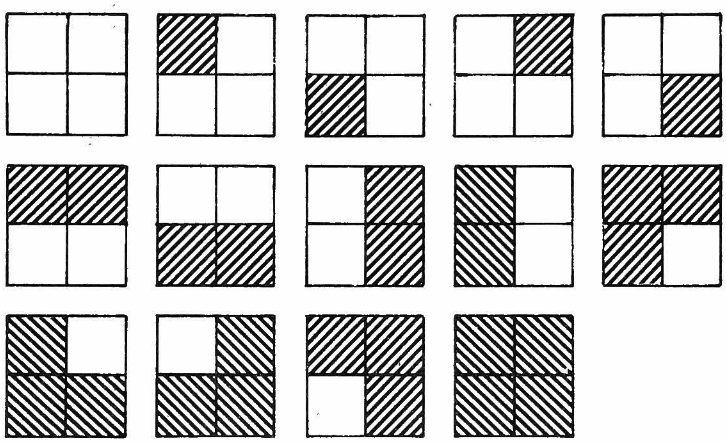
8th bit



9th bit

Appendix-B Integer sequences corresponding to legal patterns and supplementary results of SYM-pictures and LSC-pictures

* Legal patterns (m = 2)



* Integer sequences b(i)

MESH SIZE = 3 number = 9

```
1 0 0
2 0 0
1 1 0
3 0 0
2 1 0
1 1 1
3 1 0
2 2 0
2 1 1
```

MESH SIZE = 4 number = 38

```
1 0 0 0
2 0 0 0
1 1 0 0
3 0 0 0
2 1 0 0
1 1 1 0
4 0 0 0
3 1 0 0
2 2 0 0
2 1 1 0
1 1 1 1
4 1 0 0
3 2 0 0
3 1 1 0
2 2 1 0
2 1 1 1
4 2 0 0
4 1 1 0
3 3 0 0
3 2 1 0
3 1 1 1
2 2 2 0
2 2 1 1
4 3 0 0
4 2 1 0
4 1 1 1
3 3 1 0
3 2 2 0
3 2 1 1
2 2 2 1
4 4 0 0
4 3 1 0
4 2 2 0
4 2 1 1
3 3 2 0
3 3 1 1
3 2 2 1
2 2 2 2
```

MESH SIZE = 5 number = 125

```
1 0 0 0 0
2 0 0 0 0
1 1 0 0 0
3 0 0 0 0
2 1 0 0 0
1 1 1 0 0
4 0 0 0 0
3 1 0 0 0
2 2 0 0 0
2 1 1 0 0
1 1 1 1 0
5 0 0 0 0
4 1 0 0 0
3 2 0 0 0
3 1 1 0 0
2 2 1 0 0
2 1 1 1 0
1 1 1 1 1
5 1 0 0 0
4 2 0 0 0
4 1 1 0 0
3 3 0 0 0
3 2 1 0 0
3 1 1 1 0
2 2 2 0 0
2 2 1 1 0
2 1 1 1 1
5 2 0 0 0
5 1 1 0 0
4 3 0 0 0
4 2 1 0 0
4 1 1 1 0
3 3 1 0 0
3 2 2 0 0
3 2 1 1 0
3 1 1 1 1
2 2 2 1 0
2 2 1 1 1
5 3 0 0 0
5 2 1 0 0
5 1 1 1 0
4 4 0 0 0
4 3 1 0 0
4 2 2 0 0
4 2 1 1 0
4 1 1 1 1
3 3 2 0 0
3 3 1 1 0
3 2 2 1 0
3 2 1 1 1
2 2 2 2 0
2 2 2 1 1
5 4 0 0 0
5 3 1 0 0
5 2 2 0 0
5 2 1 1 0
5 1 1 1 1
4 4 1 0 0
4 3 2 0 0
4 3 1 1 0
4 2 2 1 0
4 2 1 1 1
3 3 3 0 0
3 3 2 1 0
3 3 1 1 1
3 2 2 2 0
3 2 2 1 1
2 2 2 2 1
5 5 0 0 0
5 4 1 0 0
5 3 2 0 0
5 3 1 1 0
5 2 2 1 0
5 2 1 1 1
4 4 2 0 0
4 4 1 1 0
4 3 3 0 0
4 3 2 1 0
4 3 1 1 1
4 2 2 2 0
4 2 2 1 1
3 3 3 1 0
3 3 2 2 0
3 3 2 1 1
3 2 2 2 1
2 2 2 2 2
5 5 1 0 0
5 4 2 0 0
5 4 1 1 0
5 3 3 0 0
5 3 2 1 0
5 3 1 1 1
5 2 2 2 0
5 2 2 1 1
4 4 3 0 0
4 4 2 1 0
4 4 1 1 1
4 3 3 1 0
4 3 2 2 0
4 3 2 1 1
4 2 2 2 1
3 3 3 2 0
3 3 3 1 1
3 3 2 2 1
3 2 2 2 2
5 5 2 0 0
5 5 1 1 0
5 4 3 0 0
5 4 2 1 0
5 4 1 1 1
5 3 3 1 0
5 3 2 2 0
5 3 2 1 1
5 2 2 2 1
4 4 4 0 0
4 4 3 1 0
4 4 2 2 0
4 4 2 1 1
4 3 3 2 0
4 3 3 1 1
4 3 2 2 1
4 2 2 2 2
3 3 3 3 0
3 3 3 2 1
3 3 2 2 2
```

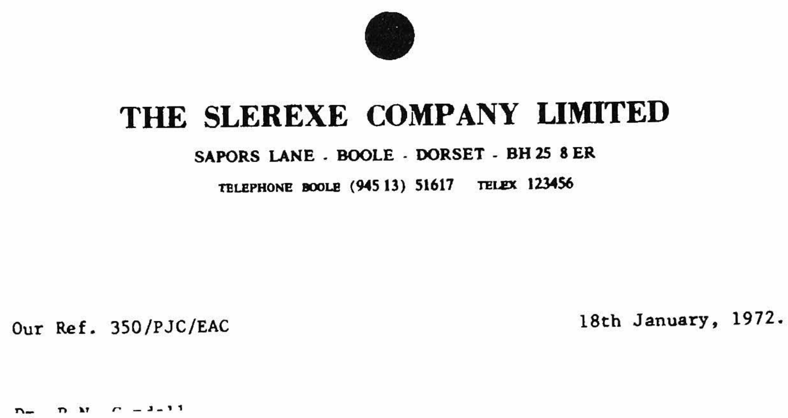
MESH SIZE = 6 number = 490

1 0 0 0 0 0	4 2 1 1 0 0	4 2 2 1 1 0	6 2 2 1 1 0	5 4 1 1 1 1
2 0 0 0 0 0	4 1 1 1 1 0	4 2 1 1 1 1	6 2 1 1 1 1	5 3 3 2 0 0
1 1 0 0 0 0	3 3 2 0 0 0	3 3 3 1 0 0	5 5 2 0 0 0	5 3 3 1 1 0
3 0 0 0 0 0	3 3 1 1 0 0	3 3 2 2 0 0	5 5 1 1 0 0	5 3 2 2 1 0
2 1 0 0 0 0	3 2 2 1 0 0	3 3 2 1 1 0	5 4 3 0 0 0	5 3 2 1 1 1
1 1 1 0 0 0	3 2 1 1 1 0	3 3 1 1 1 1	5 4 2 1 0 0	5 2 2 2 2 0
4 0 0 0 0 0	3 1 1 1 1 1	3 2 2 2 1 0	5 4 1 1 1 0	5 2 2 2 1 1
3 1 0 0 0 0	2 2 2 2 0 0	3 2 2 1 1 1	5 3 3 1 0 0	4 4 4 1 0 0
2 2 0 0 0 0	2 2 2 1 1 0	2 2 2 2 2 0	5 3 2 2 0 0	4 4 3 2 0 0
2 1 1 0 0 0	2 2 1 1 1 1	2 2 2 2 1 1	5 3 2 1 1 0	4 4 3 1 1 0
1 1 1 1 0 0	6 3 0 0 0 0	6 5 0 0 0 0	5 3 1 1 1 1	4 4 2 2 1 0
5 0 0 0 0 0	6 2 1 0 0 0	6 4 1 0 0 0	5 2 2 2 1 0	4 4 2 1 1 1
4 1 0 0 0 0	6 1 1 1 0 0	6 3 2 0 0 0	5 2 2 1 1 1	4 3 3 3 0 0
3 2 0 0 0 0	5 4 0 0 0 0	6 3 1 1 0 0	4 4 4 0 0 0	4 3 3 2 1 0
3 1 1 0 0 0	5 3 1 0 0 0	6 2 2 1 0 0	4 4 3 1 0 0	4 3 3 1 1 1
2 2 1 0 0 0	5 2 2 0 0 0	6 2 1 1 1 0	4 4 2 2 0 0	4 3 2 2 2 0
2 1 1 1 0 0	5 2 1 1 0 0	6 1 1 1 1 1	4 4 2 1 1 0	4 3 2 2 1 1
1 1 1 1 1 0	5 1 1 1 1 0	5 5 1 0 0 0	4 4 1 1 1 1	4 2 2 2 2 1
6 0 0 0 0 0	4 4 1 0 0 0	5 4 2 0 0 0	4 3 3 2 0 0	3 3 3 3 1 0
5 1 0 0 0 0	4 3 2 0 0 0	5 4 1 1 0 0	4 3 3 1 1 0	3 3 3 2 2 0
4 2 0 0 0 0	4 3 1 1 0 0	5 3 3 0 0 0	4 3 2 2 1 0	3 3 3 2 1 1
4 1 1 0 0 0	4 2 2 1 0 0	5 3 2 1 0 0	4 3 2 1 1 1	3 3 2 2 2 1
3 3 0 0 0 0	4 2 1 1 1 0	5 3 1 1 1 0	4 2 2 2 2 0	3 2 2 2 2 2
3 2 1 0 0 0	4 1 1 1 1 1	5 2 2 2 0 0	4 2 2 2 1 1	6 6 2 0 0 0
3 1 1 1 0 0	3 3 3 0 0 0	5 2 2 1 1 0	3 3 3 3 0 0	6 6 1 1 0 0
2 2 2 0 0 0	3 3 2 1 0 0	5 2 1 1 1 1	3 3 3 2 1 0	6 5 3 0 0 0
2 2 1 1 0 0	3 3 1 1 1 0	4 4 3 0 0 0	3 3 3 1 1 1	6 5 2 1 0 0
2 1 1 1 1 0	3 2 2 2 0 0	4 4 2 1 0 0	3 3 2 2 2 0	6 5 1 1 1 0
1 1 1 1 1 1	3 2 2 1 1 0	4 4 1 1 1 0	3 3 2 2 1 1	6 4 4 0 0 0
6 1 0 0 0 0	3 2 1 1 1 1	4 3 3 1 0 0	3 2 2 2 2 1	6 4 3 1 0 0
5 2 0 0 0 0	2 2 2 2 1 0	4 3 2 2 0 0	2 2 2 2 2 2	6 4 2 2 0 0
5 1 1 0 0 0	2 2 2 1 1 1	4 3 2 1 1 0	6 6 1 0 0 0	6 4 2 1 1 0
4 3 0 0 0 0	6 4 0 0 0 0	4 3 1 1 1 1	6 5 2 0 0 0	6 4 1 1 1 1
4 2 1 0 0 0	6 3 1 0 0 0	4 2 2 2 1 0	6 5 1 1 0 0	6 3 3 2 0 0
4 1 1 1 0 0	6 2 2 0 0 0	4 2 2 1 1 1	6 4 3 0 0 0	6 3 3 1 1 0
3 3 1 0 0 0	6 2 1 1 0 0	3 3 3 2 0 0	6 4 2 1 0 0	6 3 2 2 1 0
3 2 2 0 0 0	6 1 1 1 1 0	3 3 3 1 1 0	6 4 1 1 1 0	6 3 2 1 1 1
3 2 1 1 0 0	5 5 0 0 0 0	3 3 2 2 1 0	6 3 3 1 0 0	6 2 2 2 2 0
3 1 1 1 1 0	5 4 1 0 0 0	3 3 2 1 1 1	6 3 2 2 0 0	6 2 2 2 1 1
2 2 2 1 0 0	5 3 2 0 0 0	3 2 2 2 2 0	6 3 2 1 1 0	5 5 4 0 0 0
2 2 1 1 1 0	5 3 1 1 0 0	3 2 2 2 1 1	6 3 1 1 1 1	5 5 3 1 0 0
2 1 1 1 1 1	5 2 2 1 0 0	2 2 2 2 2 1	6 2 2 2 1 0	5 5 2 2 0 0
6 2 0 0 0 0	5 2 1 1 1 0	6 6 0 0 0 0	6 2 2 1 1 1	5 5 2 1 1 0
6 1 1 0 0 0	5 1 1 1 1 1	6 5 1 0 0 0	5 5 3 0 0 0	5 5 1 1 1 1
5 3 0 0 0 0	4 4 2 0 0 0	6 4 2 0 0 0	5 5 2 1 0 0	5 4 4 1 0 0
5 2 1 0 0 0	4 4 1 1 0 0	6 4 1 1 0 0	5 5 1 1 1 0	5 4 3 2 0 0
5 1 1 1 0 0	4 3 3 0 0 0	6 3 3 0 0 0	5 4 4 0 0 0	5 4 3 1 1 0
4 4 0 0 0 0	4 3 2 1 0 0	6 3 2 1 0 0	5 4 3 1 0 0	5 4 2 2 1 0
4 3 1 0 0 0	4 3 1 1 1 0	6 3 1 1 1 0	5 4 2 2 0 0	5 4 2 1 1 1
4 2 2 0 0 0	4 2 2 2 0 0	6 2 2 2 0 0	5 4 2 1 1 0	5 3 3 3 0 0

5 3 3 2 1 0	5 4 3 3 0 0	5 5 4 2 0 0	6 4 3 3 1 0	6 5 5 1 1 0
5 3 3 1 1 1	5 4 3 2 1 0	5 5 4 1 1 0	6 4 3 2 2 0	6 5 4 3 0 0
5 3 2 2 2 0	5 4 3 1 1 1	5 5 3 3 0 0	6 4 3 2 1 1	6 5 4 2 1 0
5 3 2 2 1 1	5 4 2 2 2 0	5 5 3 2 1 0	6 4 2 2 2 1	6 5 4 1 1 1
5 2 2 2 2 1	5 4 2 2 1 1	5 5 3 1 1 1	6 3 3 3 2 0	6 5 3 3 1 0
4 4 4 2 0 0	5 3 3 3 1 0	5 5 2 2 2 0	6 3 3 3 1 1	6 5 3 2 2 0
4 4 4 1 1 0	5 3 3 2 2 0	5 5 2 2 1 1	6 3 3 2 2 1	6 5 3 2 1 1
4 4 3 3 0 0	5 3 3 2 1 1	5 4 4 3 0 0	6 3 2 2 2 2	6 5 2 2 2 1
4 4 3 2 1 0	5 3 2 2 2 1	5 4 4 2 1 0	5 5 5 2 0 0	6 4 4 4 1 0
4 4 3 1 1 1	5 2 2 2 2 2	5 4 4 1 1 1	5 5 5 1 1 0	6 4 4 3 1 0
4 4 2 2 2 0	4 4 4 3 0 0	5 4 3 3 1 0	5 5 4 3 0 0	6 4 4 2 2 0
4 4 2 2 1 1	4 4 4 2 1 0	5 4 3 2 2 0	5 5 4 2 1 0	6 4 4 2 1 1
4 3 3 3 1 0	4 4 4 1 1 1	5 4 3 2 1 1	5 5 4 1 1 1	6 4 3 3 2 0
4 3 3 2 2 0	4 4 3 3 1 0	5 4 2 2 2 1	5 5 3 3 1 0	6 4 3 3 1 1
4 3 3 2 1 1	4 4 3 2 2 0	5 3 3 3 2 0	5 5 3 2 2 0	6 4 3 2 2 1
4 3 2 2 2 1	4 4 3 2 1 1	5 3 3 3 1 1	5 5 3 2 1 1	6 4 2 2 2 2
4 2 2 2 2 2	4 4 2 2 2 1	5 3 3 2 2 1	5 5 2 2 2 1	6 3 3 3 3 0
3 3 3 3 2 0	4 3 3 3 2 0	5 3 2 2 2 2	5 4 4 4 0 0	6 3 3 3 2 1
3 3 3 3 1 1	4 3 3 3 1 1	4 4 4 4 0 0	5 4 4 3 1 0	6 3 3 2 2 2
3 3 3 2 2 1	4 3 3 2 2 1	4 4 4 3 1 0	5 4 4 2 2 0	5 5 5 3 0 0
3 3 2 2 2 2	4 3 2 2 2 2	4 4 4 2 2 0	5 4 4 2 1 1	5 5 5 2 1 0
6 6 3 0 0 0	3 3 3 3 3 0	4 4 4 2 1 1	5 4 3 3 2 0	5 5 5 1 1 1
6 6 2 1 0 0	3 3 3 3 2 1	4 4 3 3 2 0	5 4 3 3 1 1	5 5 4 4 0 0
6 6 1 1 1 0	3 3 3 2 2 2	4 4 3 3 1 1	5 4 3 2 2 1	5 5 4 3 1 0
6 5 4 0 0 0	6 6 4 0 0 0	4 4 3 2 2 1	5 4 2 2 2 2	5 5 4 2 2 0
6 5 3 1 0 0	6 6 3 1 0 0	4 4 2 2 2 2	5 3 3 3 3 0	5 5 4 2 1 1
6 5 2 2 0 0	6 6 2 2 0 0	4 3 3 3 3 0	5 3 3 3 2 1	5 5 3 3 2 0
6 5 2 1 1 0	6 6 2 1 1 0	4 3 3 3 2 1	5 3 3 2 2 2	5 5 3 3 1 1
6 5 1 1 1 1	6 6 1 1 1 1	4 3 3 2 2 2	4 4 4 4 1 0	5 5 3 2 2 1
6 4 4 1 0 0	6 5 5 0 0 0	3 3 3 3 3 1	4 4 4 3 2 0	5 5 2 2 2 2
6 4 3 2 0 0	6 5 4 1 0 0	3 3 3 3 2 2	4 4 4 3 1 1	5 4 4 4 1 0
6 4 3 1 1 0	6 5 3 2 0 0	6 6 5 0 0 0	4 4 4 2 2 1	5 4 4 3 2 0
6 4 2 2 1 0	6 5 3 1 1 0	6 6 4 1 0 0	4 4 3 3 3 0	5 4 4 3 1 1
6 4 2 1 1 1	6 5 2 2 1 0	6 6 3 2 0 0	4 4 3 3 2 1	5 4 4 2 2 1
6 3 3 3 0 0	6 5 2 1 1 1	6 6 3 1 1 0	4 4 3 2 2 2	5 4 3 3 3 0
6 3 3 2 1 0	6 4 4 2 0 0	6 6 2 2 1 0	4 3 3 3 3 1	5 4 3 3 2 1
6 3 3 1 1 1	6 4 4 1 1 0	6 6 2 1 1 1	4 3 3 3 2 2	5 4 3 2 2 2
6 3 2 2 2 0	6 4 3 3 0 0	6 5 5 1 0 0	3 3 3 3 3 2	5 3 3 3 3 1
6 3 2 2 1 1	6 4 3 2 1 0	6 5 4 2 0 0	6 6 6 0 0 0	5 3 3 3 2 2
6 2 2 2 2 1	6 4 3 1 1 1	6 5 4 1 1 0	6 6 5 1 0 0	4 4 4 4 2 0
5 5 5 0 0 0	6 4 2 2 2 0	6 5 3 3 0 0	6 6 4 2 0 0	4 4 4 4 1 1
5 5 4 1 0 0	6 4 2 2 1 1	6 5 3 2 1 0	6 6 4 1 1 0	4 4 4 3 3 0
5 5 3 2 0 0	6 3 3 3 1 0	6 5 3 1 1 1	6 6 3 3 0 0	4 4 4 3 2 1
5 5 3 1 1 0	6 3 3 2 2 0	6 5 2 2 2 0	6 6 3 2 1 0	4 4 4 2 2 2
5 5 2 2 1 0	6 3 3 2 1 1	6 5 2 2 1 1	6 6 3 1 1 1	4 4 3 3 3 1
5 5 2 1 1 1	6 3 2 2 2 1	6 4 4 3 0 0	6 6 2 2 2 0	4 4 3 3 2 2
5 4 4 2 0 0	6 2 2 2 2 2	6 4 4 2 1 0	6 6 2 2 1 1	4 3 3 3 3 2
5 4 4 1 1 0	5 5 5 1 0 0	6 4 4 1 1 1	6 5 5 2 0 0	3 3 3 3 3 3

* Supplementary results of SYM-pictures and LSC-pictures

CCITT NO.1



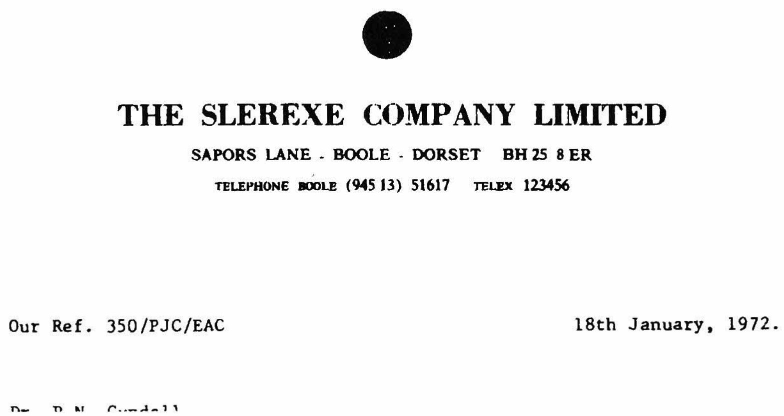
THE SLEREXE COMPANY LIMITED

SAPORS LANE - BOOLE - DORSET - BH 25 8 ER

TELEPHONE BOOLE (945 13) 51617 TELEX 123456

Our Ref. 350/PJC/EAC 18th January, 1972.

original image



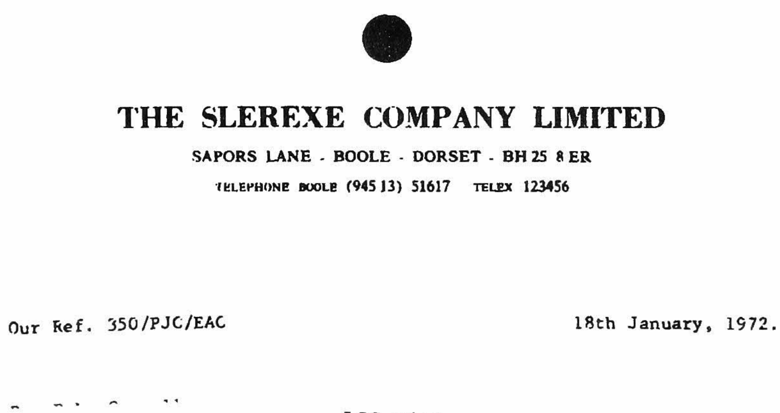
THE SLEREXE COMPANY LIMITED

SAPORS LANE - BOOLE - DORSET BH 25 8 ER

TELEPHONE BOOLE (945 13) 51617 TELEX 123456

Our Ref. 350/PJC/EAC 18th January, 1972.

SYM-picture



THE SLEREXE COMPANY LIMITED

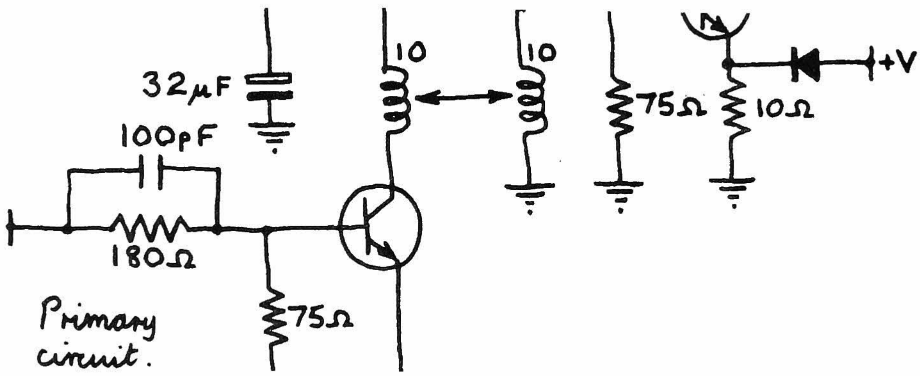
SAPORS LANE - BOOLE - DORSET - BH 25 8 ER

TELEPHONE BOOLE (945 13) 51617 TELEX 123456

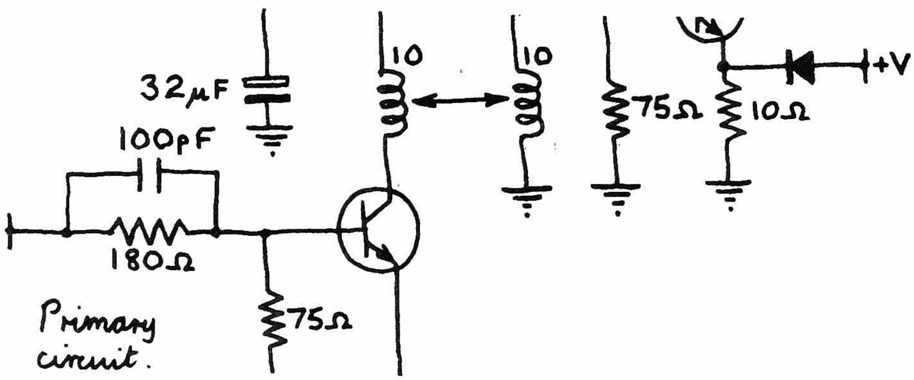
Our Ref. 350/PJC/EAC 18th January, 1972.

LSC-picture

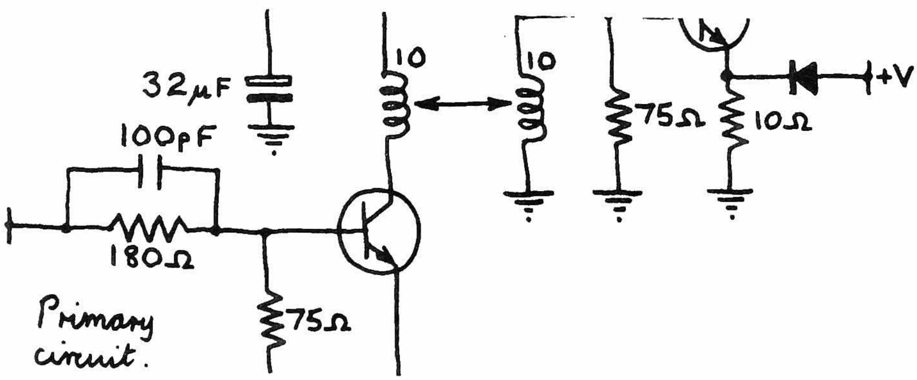
CCITT NO.2



original image



SYM-picture



LSC-picture

CCITT NO.4

- 34

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications.

original image

- 34

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications.

SYM-picture

- 34

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications.

LSC-picture

$$T_R = T_0 + (f_0 - f)\,\frac{T}{\Delta f} \quad (\text{avec } T_0 > T)$$

(voir fig. 4).

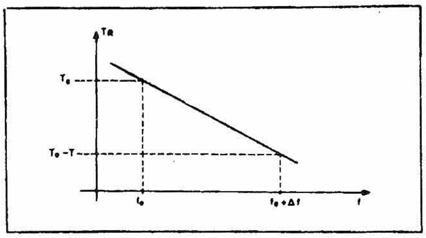


FIG. 4

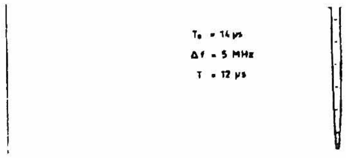


FIG. 5

On saisit physiquement le phénomène de compression en réalisant que lorsque le signal $S(t)$ entre dans la ligne à retard (LAR) la fréquence qui entre la première à l'instant 0 est la fréquence basse f_0, qui met un temps T_0 pour traverser. La fréquence f entre à l'instant $t = (f - f_0)\,\frac{T}{\Delta f}$ et elle met un temps $T_0 - (f - f_0)\,\frac{T}{}$ pour traverser, ce qui la fait ressortir

original image

$$T_R = T_0 + (f_0 - f)\,\frac{T}{\Delta f} \quad (\text{avec } T_0 > T)$$

(voir fig. 4).

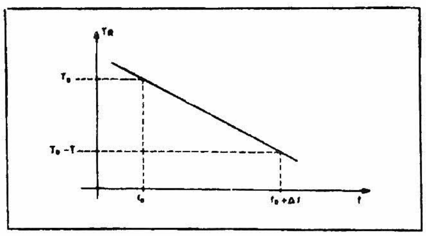


FIG. 4

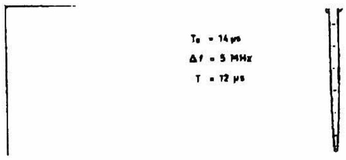


FIG. 5

On saisit physiquement le phénomène de compression en réalisant que lorsque le signal $S(t)$ entre dans la ligne à retard (LAR) la fréquence qui entre la première à l'instant 0 est la fréquence basse f_0, qui met un temps T_0 pour traverser. La fréquence f entre à l'instant $t = (f - f_0)\,\frac{T}{\Delta f}$ et elle met un temps $T_0 - (f - f_0)\,\frac{T}{}$ pour traverser, ce qui la fait ressortir

SYM-picture

$$T_R = T_0 + (f_0 - f)\,\frac{T}{\Delta f} \quad (\text{avec } T_0 > T)$$

(voir fig. 4).

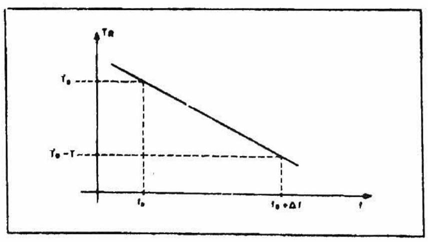


FIG. 4

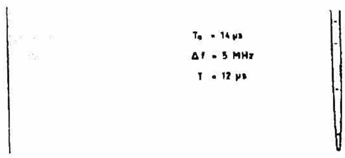


FIG. 5

On saisit physiquement le phénomène de compression en réalisant que lorsque le signal $S(t)$ entre dans la ligne à retard (LAR) la fréquence qui entre la première à l'instant 0 est la fréquence basse f_0, qui met un temps T_0 pour traverser. La fréquence f entre à l'instant $t = (f - f_0)\,\frac{T}{\Delta f}$ et elle met un temps $T_0 - (f - f_0)\,\frac{T}{}$ pour traverser, ce qui la fait ressortir

LSC-picture

Appendix-C Complete code words

* Zero memory coding (ZMC)

BLOCK CODE TABLE

NUMBER OF LEGAL PATTERNS	CODE WORD
1	1
2	011
3	0101111
4	001100
5	0101110
6	001010
7	001001
8	0101011
9	001101
10	001000
11	0100100
12	0100011
13	0011111
14	01011001
15	0100001
16	01010100
17	01011000
18	01010001
19	00111101
20	01000100
21	00111011
22	01000000
23	00111000
24	00101100
25	00101101
26	00101111
27	010110110
28	010110101
29	010100100
30	010101010
31	010101011
32	010010101
33	010100111
34	010011110
35	010100110
36	010001010
37	010011111
38	010011100
39	010010100
40	010100001
41	010011101
42	010001011
43	010011010
44	010011000
45	010100101
46	001110010
47	010010110
48	0101101111
49	010100000
50	001111000
51	010011001
52	010011011
53	010000010
54	010010111
55	001110101
56	001011100
57	001011101
58	001111001
59	001110100
60	0101101110
61	001110011
62	0100000111
63	0101101000
64	0100000110
65	01011010011
66	01011010010
EOL	00000000001

* One dimensional coding (ODC)

code words for W patterns

* terminating code

run length	code word	run length	code word
0	00110101	32	00011011
1	1111	33	00010010
2	0111	34	00010011
3	1000	35	00010100
4	1011	36	00010101
5	1100	37	00010110
6	1110	38	00010111
7	10011	39	00101000
8	10100	40	00101001
9	00111	41	00101010
10	01000	42	00101011
11	000111	43	00101100
12	001000	44	00101101
13	000011	45	00000100
14	110100	46	00000101
15	110101	47	00001010
16	101010	48	00001011
17	101011	49	01010010
18	0100111	50	01010011
19	0001100	51	01010100
20	0001000	52	01010101
21	0010111	53	00100100
22	0000011	54	00100101
23	0000100	55	01011000
24	0101000	56	01011001
25	0101011	57	01011010
26	0010011	58	01011011
27	0100100	59	01001010
28	0011000	60	01001011
29	00000010	61	00110010
30	00000011	62	00110011
31	00011010	63	00110100

* make up code

run length	code word
64	11011
128	10010
192	010111
256	0110111
320	00110110
384	00110111
448	01100100
512	01100101
576	011000
EOL	000000000001

* Two dimensional coding (TDC)

pass mode	0001
vertical mode $V_R(1)$	010
vertical mode $V(0)$	1
Vertical mode $V_L(1)$	011
Horizontal mode	001

* Code words for legal connections of black legal patterns

present legal pattern number | code length | code word | previous legal pattern number

LEGAL PATTERN NUMBER = 1

present legal pattern number	code length	code word
1	1	1
2	4	0001
3	7	0110100
4	6	010010
5	8	01110100
7	4	0011
8	8	01101101
9	4	0010
10	7	0111111
12	5	01100
13	5	00001
14	9	000000111
16	7	0111100
17	7	0111110
18	8	00000010
19	9	011011000
21	10	0110110011
22	6	010111
23	6	010100
24	6	010110
25	6	010001
27	10	0111101110
28	7	0110101
31	7	0111011
32	8	01000001
33	6	000001
34	7	0110111
36	9	010011010
37	7	0111001
38	10	0000001100
39	8	01010110
40	10	0111101111
42	10	0000001101
43	7	0111000
44	8	01010111
48	8	01111010
49	7	0101010
50	7	0100001
52	8	01001100
53	8	01110101
57	9	011110110
59	11	01101100101
60	12	011011001000
61	7	0100111
63	8	00000001
64	9	010011011
66	12	011011001001
67	12	000000000001

LEGAL PATTERN NUMBER = 2

1	2	01
2	2	11
3	6	101111
4	7	1011000
5	5	10100
6	4	1000
8	5	00110
9	9	001111111
10	6	100101
11	5	00011
12	11	10110011111
13	10	1011001110
14	6	101010
15	5	00010
18	5	00100
19	8	10110010
20	6	000001
21	8	00001110
23	11	10110011110
25	11	00001111101
26	6	100100
27	7	0010111
29	6	000010
30	6	100111
35	6	101110
36	9	001111110
38	8	10110110
39	9	101100110
40	8	10110111
41	6	001110
42	8	00101011
45	7	0011110
46	8	10101111
47	7	1001101
51	7	0010110
52	9	000011110
54	7	1001100
55	7	1010110
56	7	1011010
57	11	00001111100
58	8	10101110
59	7	0010100
60	8	00101010
62	7	0000110
64	11	00001111111
65	8	00111110
66	11	00001111110
67	12	000000000001

LEGAL PATTERN NUMBER = 3

1	2	01
2	5	10110
3	2	11
4	6	000001
5	7	1000111
9	10	1000110101
12	9	000011111
13	8	10001100
14	5	10010
16	5	10111
17	5	10100
18	8	00001110
21	6	000010
23	7	0000110
25	9	000011110
26	3	001
27	6	100010
38	5	10011
39	6	101011
40	10	1000110100
45	6	101010
54	6	100001
57	10	1000110111
60	4	0001
64	10	1000110110
67	12	000000000001

LEGAL PATTERN NUMBER = 4

1	2	01
2	5	10010
3	6	100110
4	2	11
5	8	00000111
9	8	00001111
12	7	1001110
13	7	0000010
14	8	00001110
15	3	101
17	4	0001
18	10	1001111001
21	3	001
23	6	000010
25	8	00000110
27	10	1001111000
28	5	10001
38	9	100111111
39	11	10011110111
40	11	10011110110
44	9	100111110
48	6	100001
57	7	0000001
60	10	1001111010
64	7	0000110
67	12	000000000001

LEGAL PATTERN NUMBER = 5

1	1	1
7	4	0100
9	3	001
12	5	01101
13	4	0001
22	7	0000001
23	5	01111
24	5	01010
25	6	000011
33	6	010111
34	8	00000001
37	6	000010
43	7	0000011
49	5	01100
50	5	01110
61	7	0000010
63	6	010110

LEGAL PATTERN NUMBER = 6

1	1	1
2	5	00101
3	9	011111001
4	7	0000111
5	9	011111000
7	4	0101
8	9	011111011
9	4	0110
10	9	011111010
12	5	01000
13	4	0001
14	9	011100101
16	8	01111111
17	8	01111110
18	9	011100100
19	9	011100111
21	9	011100110
22	6	000001
23	5	00110
24	6	000010
25	6	010010
27	9	011100001
28	6	001110
31	8	01111001
32	10	0111000001
33	8	01111000
34	8	01111011
36	9	011100011
37	9	011100010
38	9	011101101
39	8	01111010
40	9	011101100
42	9	011101111
43	7	0011111
44	9	011101110
48	9	011101001
49	7	0100110
50	5	00100
52	9	011101000
53	9	011101011
57	9	011101010
59	9	010011101
60	9	010011100
61	9	010011111
63	7	0011110
64	7	0000110
66	9	010011110
67	12	000000000001

LEGAL PATTERN NUMBER = 7

1	3	001
2	3	101
3	9	000010010
4	9	000010011
5	4	1111
6	2	01
8	9	111011010
9	7	1110101
10	9	111011011
11	5	00010
12	9	111011000
13	9	111011001
14	5	11000
15	5	11001
18	6	100101
19	9	111011110
20	6	000110
21	10	0000100001
23	9	000010001
25	9	111011111
26	5	10011
27	6	110100
29	9	111011100
30	5	11011
35	6	100001
36	10	1000000001
38	9	111011101
39	9	111010010
40	7	0000101
41	7	1001000
42	9	111010011
45	7	1110001
46	7	1001001
47	6	110101
51	6	000001
52	10	1110100001
54	7	1000001
55	5	10001
56	8	11100001
57	9	111010001
58	6	111001
59	10	1000000100
60	10	1000000101
62	6	000011
64	9	100000001
65	6	000111
66	9	100000011
67	12	000000000001

LEGAL PATTERN NUMBER =

1	3	001
2	4	1001
5	6	111001
6	2	01
9	9	000010001
10	6	111000
11	4	1111
12	10	0000100001
13	9	000010011
14	7	0000110
16	5	10101
18	4	1000
19	6	000001
20	4	1101
23	8	00001010
25	9	000010010
27	10	0000101111
29	4	1100
36	5	11101
40	10	0000101110
42	4	1011
52	7	0000111
56	5	10100
59	4	0001
66	9	000010110
67	12	000000000001

LEGAL PATTERN NUMBER = 9

2	3	011
5	1	1
6	3	001
14	6	010111
18	6	000001
27	4	0100
30	7	0000111
35	7	0000110
40	5	00011
41	7	0101011
46	7	0000100
47	6	010110
51	5	00010
55	6	010100
58	7	0101010
62	8	00001011
65	8	00001010

LEGAL PATTERN NUMBER = 10

1	2	01
2	5	00010
5	10	1100101101
8	5	00001
9	10	1100101100
10	2	10
11	3	111
12	9	110111110
13	8	11011110
14	10	1100101111
16	6	000001
18	7	1100100
19	3	001
23	10	1100101110
25	11	11011111111
27	11	11011111110
31	5	00011
32	5	11010
36	8	11001010
40	10	1101111110
42	7	1101110
52	5	11000
53	6	110110
59	7	1100111
66	7	1100110
67	12	000000000001

LEGAL PATTERN NUMBER = 11

1	1	1
2	8	00001111
3	7	0110011
4	8	00001110
5	8	00001001
7	5	01011
8	7	0110010
9	5	01110
10	5	00110
12	4	0100
13	6	011000
14	8	00001000
16	7	0000110
17	7	0000001
18	8	00000001
19	8	00001011
21	7	0000011
22	5	01010
23	7	0011111
24	7	0000010
25	6	011011
27	8	00001010
28	8	00100101
31	6	011010
32	8	00100100
33	6	001110
34	8	00100111
36	8	00100110
37	5	00010
38	8	00100001
39	9	001000001
40	8	00100011
42	8	00100010
43	7	0011110
44	8	00101101
48	7	0001101
49	5	01111
50	8	00101100
52	7	0001100
53	8	00101111
57	8	00101110
59	8	00101001
60	8	00101000
61	7	0001111
63	7	0001110
64	8	00101011
66	8	00101010
67	12	000000000001

LFGAL PATTERN NUMBER = 12

2	3	011
5	5	10111
8	4	1110
10	3	110
11	3	001
14	7	1000011
16	4	0001
18	3	010
19	5	11110
27	6	100011
31	5	10011
32	4	1010
36	6	111110
40	6	111111
42	7	1000010
52	5	10110
53	7	1000001
59	5	10010
66	6	100010

LEGAL PATTERN NUMBER = 13

2	3	100
3	4	1101
4	2	01
5	5	11001
14	5	11101
15	4	0011
17	3	101
18	8	11000011
21	4	1111
27	8	11000001
28	4	0001
38	8	11000010
39	5	00101
40	7	1100010
44	5	00001
48	5	00100
57	6	111001
60	7	1100011
64	6	111000

LEGAL PATTERN NUMBER = 14

1	2	01
8	3	111
9	9	101011111
10	3	100
11	3	110
12	9	101011110
13	9	101010111
16	3	001
19	7	1010100
20	4	0001
23	7	1010110
25	9	101010110
29	5	10100
36	5	10110
42	5	10111
52	8	10101110
56	5	00001
59	6	000001
66	8	10101010

LEGAL PATTERN NUMBER = 15

1	1	1
2	8	01001100
3	6	001110
4	7	0000001
5	8	01001101
7	5	00001
8	8	01000010
9	6	001010
10	7	0010011
12	6	000001
13	4	0001
14	8	01000011
16	9	010000001
17	8	01000001
18	8	01000110
19	8	01000111
21	8	01000100
22	8	01000101
23	6	001011
24	3	011
25	8	01011010
27	8	01011011
28	8	00100001
31	8	01011000
32	8	01011001
33	8	01011110
34	6	001100
36	8	01011111
37	7	0100111
38	8	01011100
39	8	01011101
40	8	01010010
42	8	01010011
43	7	0010001
44	9	010100001
48	8	01010001
49	6	001111
50	6	001101
52	8	01010110
53	8	01010111
57	8	01010100
59	8	01010101
60	9	001001000
61	7	0100100
63	9	001001001
64	7	0100101
66	8	00100101
67	12	000000000001

LEGAL PATTERN NUMBER = 16

1	3	001
2	3	110
3	4	0100
4	8	00010001
5	6	101111
6	4	1111
8	7	0000001
9	9	000100001
10	8	00010011
11	6	000011
12	8	00010010
13	8	00011101
14	4	1010
15	4	1000
18	8	00011100
19	7	0001011
20	6	000010
21	6	100101
23	8	00011111
25	8	00011110
26	5	01111
27	8	00011001
29	6	101110
30	4	0101
35	8	00011000
36	8	00011011
38	5	01110
39	6	100100
40	8	00011010
41	5	01101
42	7	0001010
45	5	10110
46	5	01100
47	8	00000101
51	8	00000100
52	8	00000111
54	5	11100
55	7	1110111
56	8	10011111
57	8	00000110
58	8	10011110
59	7	1110110
60	6	100110
62	6	111010
64	9	100111011
65	9	100111010
66	8	10011100
67	12	000000000001

LEGAL PATTERN NUMBER = 17

1	3	011
2	3	001
3	6	000010
4	9	110101101
5	6	111111
6	3	100
8	4	0001
9	8	11010001
10	7	1110100
11	4	1100
12	9	110101100
13	9	110101111
14	6	101010
15	9	110101110

18	5	10100
19	8	11101011
20	3	010
21	9	110101001
23	9	110101000
25	9	110101011
26	6	101011
27	9	110101010
29	5	11110
30	6	111110
35	6	000001
36	7	0000111
38	9	101111101
39	9	101111100
40	9	101111111
41	6	110111
42	6	101101
45	9	101111110
46	6	110110
47	7	1110111
51	6	101100
52	9	110100001
54	6	101110
55	7	0000110
56	5	11100
57	9	101111001
58	7	1110110
59	9	101111000
60	8	11010011
62	9	101111011
64	9	101111010
65	8	11010010
66	8	11101010
67	12	000000000001

LEGAL PATTERN NUMBER = 18

1	2	10
3	2	01
4	3	001
9	8	11010101
12	9	110010011
13	8	11010100
15	4	1111
17	4	1110
21	7	1101011
23	9	110010010
25	8	11001011
26	5	11011
38	6	110100
39	4	0001
45	5	11000
54	5	00001
57	8	11001010
60	6	110011
64	8	11001000

LEGAL PATTERN NUMBER = 19

1	1	1
7	5	01010
9	7	0111001
12	3	001
13	6	011111
22	4	0110
23	6	011110
24	4	0001
25	7	0111000
33	6	010111
34	7	0111011
37	7	0111010
43	6	010001
49	6	010110
50	7	0100001
61	6	010011
63	6	010010

LEGAL PATTERN NUMBER = 20

1	1	1
2	5	00111
3	8	01010101
4	6	001011
6	8	01010100
7	4	0001
8	5	00110
9	6	011010
10	8	01010111
12	8	01010110
13	5	01110
14	8	01010001
16	9	010100001
17	8	01010011
18	8	01010010
19	8	01011101
21	8	01011100
22	8	01011111
23	5	01000
24	7	0100111
25	6	001010
27	8	01011110
28	8	01011001
31	8	01011000
32	7	0100110
33	7	0111110
34	8	01011011
35	8	01011010
37	6	011110
38	8	01100101
39	8	01100100
40	8	01100111
42	8	01100110
43	8	01100001
44	9	011000001
48	8	01100011
49	5	00100
50	8	01100010
52	8	01101101
53	8	01101100
57	8	01101111
59	8	01101110
60	9	011111111
61	6	010010
63	5	00001
64	9	011111110
66	8	01111110
67	12	000000000001

LEGAL PATTERN NUMBER = 21

1	1	1
7	4	0010
9	6	000010
12	6	001110
13	3	010
22	4	0001
23	7	0000111
24	3	011
25	7	0000110
33	7	0011001
34	7	0011000
37	7	0011011
43	7	0011010
49	8	00111111
50	6	000001
61	8	00111110
63	7	0011110

LEGAL PATTERN NUMBER = 22

1	3	100
2	5	00001
5	8	11011110
8	3	101
9	8	11011111
10	2	01
11	3	001
12	8	11011100
13	7	0000010
14	8	11011101
16	5	11000
18	6	110100
19	3	111
23	10	0001100001
25	9	000110001
27	8	00011001
31	5	00010
32	5	11001
36	6	110101
40	7	0000011
42	7	1101100
52	7	0000001
53	6	000111
59	7	1101101
66	7	0001101
67	12	000000000001

LEGAL PATTERN NUMBER = 23

2	2	01
3	2	10
5	4	0010
14	3	110
17	4	0001
18	8	00111111
26	4	1110
27	5	00001
38	5	00110
39	5	11110
40	8	00111110
45	6	001110
54	7	0011110
60	5	11111

LEGAL PATTERN NUMBER = 24

1	4	0011
2	4	1100
3	4	0001
4	2	10
5	8	00000111
9	6	111010
12	6	001011
13	8	00000110
14	7	0000010
15	2	01
17	5	11100
18	7	1110111
21	4	1111
23	6	000010
25	8	00001101
27	8	00001100
28	6	110111
38	8	00001111
39	7	1110110
40	7	0000001
44	5	11010
48	5	00100
57	6	001010
60	8	00001110
64	6	110110
67	12	000000000001

LEGAL PATTERN NUMBER = 25

2	2	10
5	4	1111
6	3	010
14	6	000011
16	5	01111
18	3	110
20	4	0110
27	5	11101
29	5	01110
36	4	0001
40	3	001
42	6	000001
56	6	000010
59	5	11100

LEGAL PATTERN NUMBER = 26

1	1	1
2	8	01111101
3	5	01010
4	8	01111100
5	8	01111111
7	7	0111001
8	6	001100
9	6	001111
10	6	001110
12	7	0111000
13	4	0001
14	8	01111110
16	8	01111001
17	8	01111000
18	8	01111011
19	7	0111011
21	8	01111010
22	8	01100101
23	4	0010
24	6	010110
25	8	01100100
27	8	01100111
28	7	0111010
31	8	01100110
32	8	01100001
33	9	011000001
34	6	000011
36	6	000010
37	8	01100011
38	8	01100010
39	8	01101101
40	8	01101100
42	8	01101111
43	6	000001
44	8	01101110
48	8	01101001
49	8	01101000
50	8	01101011
52	8	01101010
53	8	01011101
57	8	01011100
59	8	01011111
60	8	01011110
61	4	0100
63	8	00110111
64	8	00110110
66	7	0011010
67	12	000000000001

LEGAL PATTERN NUMBER = 27

1	1	1
9	7	0000011
10	2	01
11	3	001
12	7	0001110
13	7	0000001
19	6	000110
23	6	000011
25	9	000111111
31	5	00010
32	9	000111110
52	6	000010
53	8	00011110
66	7	0000010

LEGAL PATTERN NUMBER = 28

1	3	100
2	4	0010
3	8	10110111
4	8	10110110
5	6	111111
6	2	01
8	6	111110
9	8	10110001
10	9	101100001
11	5	00010
12	8	10110011
13	8	10110010
14	8	10111101
15	5	00111
18	6	000111
19	8	10111100
20	4	1110
21	8	10111111
23	8	10111110
25	8	10111001
26	7	1011010
27	8	10111000
29	7	1100001
30	7	1101001
35	6	000110
36	8	10111011
38	8	10111010
39	8	11000101
40	7	1101000
41	6	000011
42	7	1101011
45	6	000010
46	6	110011
47	7	1101010
51	5	00110
52	8	11000100
54	7	1101110
55	5	11110
56	6	000001
57	8	11000111
58	6	110010
59	8	11000110
60	9	110111111
62	6	110110
64	9	110111110
65	4	1010
66	8	11011110
67	12	000000000001

LEGAL PATTERN NUMBER = 29

1	1	1
7	4	0110
9	7	0111001
12	3	001
13	4	0001
22	5	01000
23	7	0111000
24	7	0111011
25	5	01011
33	5	01010
34	7	0111010
37	7	0100111
43	7	0100110
49	6	011111
50	6	011110
61	5	00001
63	6	010010

LEGAL PATTERN NUMBER = 30

1	3	011
2	6	101111
5	6	101110
8	2	11
9	8	10101111
10	4	1000
11	5	10011
12	8	10101110
13	7	0000101
14	7	0000100
16	4	0101
18	3	001
19	7	0000111
20	5	10110
23	7	0000110
25	7	0000001
27	8	00000001
29	4	0001
36	5	10100
40	7	1010110
42	5	10010
52	7	0000011
56	6	101010
59	4	0100
66	7	0000010
67	12	000000000001

LEGAL PATTERN NUMBER = 31

1	3	111
2	4	1010
3	6	100011
4	8	10010101
5	7	0101111
6	4	1101
8	8	10010100
9	8	10010111
10	7	0101110
11	4	0011
12	8	10010110
13	8	10010001
14	5	00100
15	4	1011
18	9	100100001
19	8	10010011
20	5	00101
21	8	10010010
23	8	10011101
25	8	10011100
26	4	1100
27	8	10011111
29	6	100010
30	8	10011110
35	8	10011001
36	8	10011000
38	6	011111
39	8	10011011
40	8	10011010
41	8	10000101
42	8	10000100
45	5	01010
46	4	0001
47	6	011110
51	4	0100
52	8	10000111
54	5	00001
55	5	01110
56	8	10000110
57	8	10000001
58	9	100000001
59	8	10000011
60	8	10000010
62	4	0110
64	8	01011011
65	8	01011010
66	7	0101100
67	12	000000000001

LEGAL PATTERN NUMBER = 32

1	4	1110
2	4	0111
5	5	00111
8	2	10
9	7	1100110
10	6	110110
11	6	000011
12	8	11001111
13	8	11001110
14	6	110001
16	4	0010
18	3	010
19	7	1100001
20	4	1111
23	6	000010
25	6	110010
27	8	11011101
29	4	0001
36	6	000001
40	8	11011100
42	4	0110
52	8	11011111
56	5	11010
59	5	00110
66	8	11011110
67	12	000000000001

LEGAL PATTERN NUMBER = 33

2	3	111
5	5	11010
8	2	01
14	7	0001111
16	3	001
18	2	10
20	4	1100
27	7	0001110
29	6	110111
36	5	00001
40	6	000110
42	6	000001
56	5	00010
59	6	110110

LEGAL PATTERN NUMBER = 34

2	3	111
3	2	01
5	5	10010
14	3	101
17	3	110
18	7	1001110
26	3	001
27	5	00001
38	6	000001
39	4	0001
40	5	10001
45	7	1001111
54	6	100110
60	6	100001

LEGAL PATTERN NUMBER = 35

1	3	001
2	4	1110
3	2	10
4	3	010
5	6	110101
9	7	0000111
12	7	0000110
13	7	0000001
14	6	110100
15	6	000010
17	4	1100
18	8	00000001
21	7	0000011
23	7	0000010
25	7	0001001
26	5	11111
27	7	0001000
38	5	00011
39	3	011
40	7	0001011
45	7	1101111
54	7	1101110
57	5	11110
60	6	110110
64	7	0001010
67	12	000000000001

LEGAL PATTERN NUMBER = 36

2	2	10
5	3	001
6	4	1110
14	4	1111
18	6	010111
27	7	0000011
30	4	0001
35	3	110
40	7	0000010
41	5	01010
46	6	010110
47	4	0110
51	6	000011
55	4	0111
58	4	0100
62	7	0000001
65	6	000010

LEGAL PATTERN NUMBER = 37

2	2	01
5	4	0001
6	3	110
14	5	00001
18	7	0010101
27	5	10100
30	7	0010100
35	3	100
40	7	0010111
41	5	00100
46	6	001111
47	7	0010110
51	5	10101
55	4	1011
58	3	111
62	6	001110
65	5	00110

LEGAL PATTERN NUMBER = 38

1	2	11
4	2	01
9	7	1001111
12	7	1001110
13	7	1000111
15	3	101
21	3	001

23 6 100001
25 7 1000110
28 5 10010
44 6 100110
48 4 0001
57 5 00001
64 6 100010

LEGAL PATTERN NUMBER = 39
2 3 111
5 3 010
6 3 101
14 3 001
18 4 0001
27 6 011010
30 4 1100
35 5 00001
40 7 0110111
41 3 100
46 4 1101
47 7 0110110
51 4 0111
55 7 0110001
58 8 01100001
62 7 0110011
65 7 0110010

LEGAL PATTERN NUMBER = 40
1 1 1
4 2 01
9 7 0010011
12 7 0010010
13 7 0000101
15 4 0011
21 4 0001
23 6 001000
25 7 0000100
28 6 000001
44 7 0000111
48 7 0000110
57 6 001011
64 6 001010

LEGAL PATTERN NUMBER = 41
1 1 1
7 5 01111
9 3 001
12 4 0001
13 6 000001
22 6 000011
23 6 000010
24 6 010101
25 5 01110
33 6 010100
34 6 010111
37 6 010110

43 6 010001
49 4 0110
50 7 0100001
61 6 010011
63 6 010010

LEGAL PATTERN NUMBER = 42
1 1 1
9 8 00111111
10 5 00110
11 2 01
12 8 00111110
13 7 0000001
19 4 0010
23 6 001110
25 8 00000001
31 7 0000011
32 4 0001
52 7 0011110
53 5 00001
66 7 0000010

LEGAL PATTERN NUMBER = 43
2 2 01
5 4 1110
6 3 110
14 5 11111
18 7 1111010
27 4 0001
30 3 100
35 6 000010
40 6 000011
41 5 10111
46 3 001
47 6 000001
51 4 1010
55 6 111100
58 8 11110111
62 5 10110
65 8 11110110

LEGAL PATTERN NUMBER = 44
1 3 001
2 4 0110
3 2 11
4 5 01111
5 7 1010111
9 7 1010110
12 7 1010001
13 8 10100001
14 3 010
15 5 01110
17 4 0001
18 7 1010011
21 6 101010
23 7 1010010
25 7 1001101
26 4 1000

27 7 1001100
38 5 10010
39 5 10110
40 7 1001111
45 7 1001110
54 6 101110
57 8 10111111
60 8 10111110
64 7 1011110
67 12 000000000001

LEGAL PATTERN NUMBER = 45
1 1 1
7 5 00011
9 6 000001
12 5 00010
13 3 011
22 7 0000001
23 6 000011
24 3 001
25 6 000010
33 6 010001
34 7 0100001
37 6 010011
43 5 01010
49 6 010010
50 7 0101111
61 7 0101110
63 6 010110

LEGAL PATTERN NUMBER = 46
1 1 1
9 6 001010
10 2 01
11 5 00111
12 7 0000111
13 7 0000110
19 5 00110
23 6 000010
25 8 00101111
31 6 000001
32 5 00100
52 4 0001
53 8 00101110
66 7 0010110

LEGAL PATTERN NUMBER = 47
1 1 1
7 6 011111
9 4 0110
12 6 011110
13 5 01110
22 5 00101
23 5 00100
24 5 00111
25 5 00110
33 4 0100
34 5 00001
37 6 000001

43 5 00011
49 5 00010
50 6 010111
61 6 010110
63 5 01010

LEGAL PATTERN NUMBER = 48
2 3 101
5 3 001
6 2 01
14 7 1110111
18 7 1001101
27 6 111001
30 3 110
35 7 1001100
40 5 10010
41 4 1111
46 4 1000
47 6 111010
51 4 0001
55 7 1001111
58 7 1110110
62 6 111000
65 7 1001110

LEGAL PATTERN NUMBER = 49
2 3 001
5 3 100
6 2 01
14 5 11111
18 6 000010
27 3 101
30 8 00001111
35 6 000001
40 6 110111
41 4 0001
46 6 110110
47 4 1100
51 4 1110
55 5 11010
58 5 11110
62 8 00001110
65 7 0000110

LEGAL PATTERN NUMBER = 50
2 3 100
5 3 110
6 2 01
14 5 11111
18 6 000001
27 4 1110
30 6 101110
35 7 0000101
40 4 1010
41 3 001
46 5 11110
47 5 10110
51 6 101111

55 4 0001
58 7 0000100
62 7 0000111
65 7 0000110

LEGAL PATTERN NUMBER = 51
1 1 1
7 3 001
9 5 00010
12 5 01111
13 6 011011
22 5 01011
23 6 000111
24 6 000110
25 4 0100
33 6 011010
34 6 000001
37 5 01110
43 5 01010
49 5 01100
50 7 0000001
61 6 000011
63 6 000010

LEGAL PATTERN NUMBER = 52
2 3 100
5 4 0100
8 3 111
14 6 010101
16 4 1011
18 3 001
20 3 011
27 6 010100
29 4 1010
36 5 11001
40 5 11000
42 4 1101
56 4 0001
59 5 01011

LEGAL PATTERN NUMBER = 53
2 2 01
5 6 111110
6 4 1011
14 3 001
18 8 11111111
27 4 1001
30 8 11111110
35 3 110
40 7 1111001
41 7 1111000
46 4 1000
47 4 1110
51 7 1111011
55 4 1010
58 4 0001
62 7 1111110
65 7 1111010

LEGAL PATTERN NUMBER = 54
1 3 011
2 6 010001
3 5 11010
4 2 10
5 7 0101101
9 7 0100001
12 7 0101100
13 6 010011
14 7 0101111
15 3 001
17 4 1100
18 7 0101110
21 4 1111
23 7 0101001
25 7 0101000
27 7 0101011
28 7 0101010
38 6 010010
39 6 110110
40 8 11011111
44 8 11011110
48 4 0001
57 4 1110
60 7 1101110
64 5 00001
67 12 000000000001

LEGAL PATTERN NUMBER = 55
1 1 1
7 3 001
9 6 010011
12 6 010010
13 4 0001
22 5 01000
23 5 01011
24 5 01010
25 6 000001
33 7 0000001
34 6 000011
37 6 000010
43 4 0110
49 7 0111111
50 5 01110
61 7 0111110
63 6 011110

LEGAL PATTERN NUMBER = 56
1 2 01
2 7 0000111
5 7 0000110
8 5 10111
9 7 1001101
10 3 111
11 3 110
12 5 00010
13 7 1001100
14 7 1001111

16 7 1001110
18 7 1001001
19 4 1000
23 7 1001000
25 7 1001011
27 7 1001010
31 7 0001101
32 5 10110
36 6 000001
40 7 0001100
42 7 0001111
52 3 001
53 4 1010
59 7 0001110
66 6 000010
67 12 000000000001

LEGAL PATTERN NUMBER = 57
2 2 01
3 3 001
5 3 110
14 4 1110
17 4 1001
18 7 1011101
26 5 10001
27 4 1010
38 7 1011100
39 5 10110
40 6 100001
45 7 1011111
54 4 1111
60 7 1011110

LEGAL PATTERN NUMBER = 58
1 2 01
4 1 1
9 7 0000001
12 8 00000001
13 4 0001
15 5 00001
21 5 00110
23 7 0000011
25 7 0000010
28 6 001110
44 8 00111111
48 8 00111110
57 4 0010
64 7 0011110

LEGAL PATTERN NUMBER = 59
1 1 1
7 6 010101
9 5 00011
12 6 010100
13 6 010111
22 6 010110
23 5 00010
24 6 010001

25 4 0111
33 7 0100001
34 6 010011
37 6 010010
43 4 0110
49 6 000011
50 6 000010
61 3 001
63 6 000001

LEGAL PATTERN NUMBER = 60
1 1 1
7 4 0010
9 6 000001
12 5 01010
13 4 0001
22 7 0000001
23 3 011
24 6 000011
25 6 000010
33 6 001101
34 6 001100
37 6 001111
43 6 001110
49 7 0101111
50 7 0101110
61 4 0100
63 6 010110

LEGAL PATTERN NUMBER = 61
1 4 0111
2 3 010
3 2 11
4 7 0000101
5 7 0000100
9 7 0000111
12 7 0000110
13 7 0001101
14 5 10111
15 5 10110
17 3 001
18 7 0001100
21 6 100110
23 7 0001111
25 7 0001110
26 4 0110
27 7 0001001
38 6 100111
39 5 10010
40 7 0001000
45 4 1000
54 6 000001
57 7 0001011
60 4 1010
64 7 0001010
67 12 000000000001

LEGAL PATTERN NUMBER = 62
1 2 01

2 4 0001
5 7 1110111
8 7 1110110
9 7 1110001
10 2 10
11 3 001
12 8 11100001
13 7 1110011
14 7 1110010
16 7 1111101
18 6 111010
19 4 1101
23 7 1111100
25 7 1111111
27 7 1111110
31 7 1111001
32 6 000001
36 7 1111000
40 7 1111011
42 7 1111010
52 4 1100
53 7 0000111
59 7 0000110
66 6 000010
67 12 000000000001

LEGAL PATTERN NUMBER = 63
1 4 0101
2 3 110
5 7 1110111
8 2 10
9 7 1110110
10 6 111010
11 3 001
12 7 1110001
13 8 11100001
14 7 1110011
16 5 00001
18 4 0111
19 7 1110010
20 4 0110
23 7 1111101
25 7 1111100
27 6 000001
29 4 0001
36 7 1111111
40 7 1111110
42 7 1111001
52 7 1111000
56 7 1111011
59 4 0100
66 7 1111010
67 12 000000000001

LEGAL PATTERN NUMBER = 64
2 4 0010
5 1 1
6 5 01110
14 5 00011
18 6 000001

27 7 0000001
30 6 000011
35 6 000010
40 3 010
41 5 00111
46 7 0111101
47 7 0111100
51 5 00010
55 5 00110
58 4 0110
62 7 0111111
65 7 0111110

LEGAL PATTERN NUMBER = 65
1 2 01
2 5 11010
3 6 110111
4 2 10
5 6 110110
9 6 110001
12 7 1100001
13 5 00001
14 6 110011
15 5 11110
17 4 0001
18 6 110010
21 6 001101
23 6 001100
25 6 001111
27 6 001110
28 4 1110
38 6 001001
39 6 001000
40 6 001011
44 6 000001
48 6 001010
57 7 1111111
60 7 1111110
64 6 111110
67 12 000000000001

LEGAL PATTERN NUMBER = 66
2 5 00101
5 3 111
6 3 010
14 3 101
18 5 00100
27 3 110
30 5 00111
35 5 00110
40 5 00001
41 6 000001
46 5 00011
47 3 100
51 5 00010
55 5 01101
58 5 01100
62 5 01111
65 5 01110